

BACHELORTHESIS

Kevin Tesch

Konzeption und prototypische Umsetzung der Nutzung einer Cloud für die ADAS/AD-Entwicklung

FAKULTÄT TECHNIK UND INFORMATIK

Department Informatik

Faculty of Computer Science and Engineering

Department Computer Science

Kevin Tesch

Konzeption und prototypische Umsetzung der Nutzung einer Cloud für die ADAS/AD-Entwick- lung

Bachelorarbeit eingereicht im Rahmen der Bachelorprüfung
im Studiengang Wirtschaftsinformatik
am Department Informatik
der Fakultät Technik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer: Prof. Dr. Olaf Zukunft
Zweitgutachter: Prof. Dr. Stefan Sarstedt

Eingereicht am: 04.11.2021

Kevin Tesch

Thema der Arbeit

Konzeption und prototypische Umsetzung der Nutzung einer Cloud für die ADAS/AD-Entwicklung

Stichworte

Autonomes Fahren, Fahrerassistenzsysteme, Cloud Computing, Big Data

Kurzzusammenfassung

Fahrerassistenzsysteme und die Möglichkeit des autonomen Fahrens nehmen eine zukunftsweisende Rolle in der Entwicklung neuer Fahrzeuge ein. Das Herzstück dieser Entwicklung bilden die großen Mengen an Testdaten, welche während realer Fahrten oder innerhalb von Simulationen generiert und analysiert werden müssen. Um die Entwicklung von Systemen wie dem Tempomat, dem Spurhalteassistenten oder der Einparkhilfe zu optimieren und eine zentrale Anlaufstelle zu haben, auf welche von überall aus zugegriffen werden kann, bietet sich die Nutzung einer Cloud an. Ziel dieser Arbeit ist es, einen prototypischen Kriterienkatalog zu entwerfen, mit welchem sich die Cloud in die Entwicklung von ADAS/AD integrieren lässt.

Kevin Tesch

Title of Thesis

Conception and prototypical implementation with the use of a cloud for ADAS / AD development

Keywords

Autonomous Driving, Advanced Driver Assistance System, Cloud Computing, Big Data

Abstract

Driver assistance systems and the possibility of autonomous driving play a pioneering role in developing new vehicles. The heart of this development is the large amount of test data that has to be generated and analyzed during real journeys or within simulations. A cloud can optimize the development of cruise control, lane departure warning systems, or parking assistance and have a central point of contact accessible from anywhere. This work aims to design a prototypical catalog of criteria with which the cloud can integrate into the development of ADAS / AD.

Inhaltsverzeichnis

Abbildungsverzeichnis.....	vi
Tabellenverzeichnis.....	viii
1 Einleitung	1
1.1 Einführung	1
1.2 Zielsetzung.....	2
1.3 Aufbau der Arbeit.....	3
2 Grundlagen.....	4
2.1 Cloud Computing	5
2.1.1 Begriffe und Hintergrund	6
2.1.2 Eigenschaften von Cloud Computing	7
2.1.3 Cloud Bereitstellungsmodelle.....	8
2.1.4 Service Modelle.....	11
2.2 Advanced Driver Assistance Systems & Autonomous Driving	13
2.2.1 Überblick.....	13
2.2.2 Entwicklung.....	19
2.2.3 Stand der Technik	20
2.3 Cloud Computing im Kontext der automobilen Assistenzsysteme	22
3 Konzeption	24
3.1 Eigenschaften des Konzepts.....	24
3.2 Konzept zur Optimierung der ADAS/AD-Entwicklung	26
3.2.1 Navigation mit Echtzeitkarten	26
3.2.2 V2X am Beispiel Platooning	38
3.2.3 Smart Parking	49
4 Prototypische Umsetzung.....	57

4.1	Ziele und Kriterien.....	57
4.2	Architektur	58
4.2.1	Clouddienste in GCP.....	58
4.2.2	Testdaten	59
4.3	Durchführung	60
4.4	Ergebnisse und Bewertung.....	62
5	Fazit und Ausblick	66
5.1	Ausblick	67
	Literaturverzeichnis.....	68
	Anhang A: Datensätze zum Platooning.....	81

Abbildungsverzeichnis

Abbildung 1: Die fünf Charakteristika des Cloud-Computings [17]	7
Abbildung 2: Cloud-Betriebsmodelle [21].....	9
Abbildung 3: Übersicht der Cloud-Bereitstellungsmodelle [21]	10
Abbildung 4: Gegenüberstellung der Verantwortlichkeiten für die Servicemodelle der Cloud [24].....	11
Abbildung 5: Die fünf Stufen der Autonomie [31].....	14
Abbildung 6: Der Zuwachs an zu berücksichtigten Testfällen durch die Kombination von Parameter [40]	17
Abbildung 7: Vielfalt der Technik und Sensoren in neuen Fahrzeugen [48]	20
Abbildung 8: Volkswagen Partnerschaft mit Microsoft für die Entwicklung des autonomen Fahrens [53].....	22
Abbildung 9: Vorgehen des Konzepts	26
Abbildung 10: Kursabweichung für 0, 100 und 150ms [60].....	29
Abbildung 11: HD Map von TomTom [62]	30
Abbildung 12: Selbstheilende Karte	32
Abbildung 13: Aktualisierung der HD-Karte mithilfe der Cloud [66].....	35
Abbildung 14: Platooning auf der Autobahn [70]	39
Abbildung 15: Platooning-Architektur (Vehicle-to-Edge-to-Cloud).....	45
Abbildung 16: Smart Parking Prinzip [55].....	50
Abbildung 17: Preisveränderung in \$ für AWS und Google pro Monat [89]	54
Abbildung 18: Platooning-Testdaten	59

Abbildungsverzeichnis

Abbildung 19: Darstellung der Züge auf einer vorgegebenen Strecke	60
Abbildung 20: Entscheidungsbaum zur Eignung verschiedener Züge.....	61
Abbildung 21: SQL-Statement zur Ermittlung eines passenden Zuges	62
Abbildung 22: Ergebnis für die Suche nach einem passenden Zug	62
Abbildung 23: Preiskalkulation für a) die leichteste und b) die höchste Ausstattung [97]	65

Tabellenverzeichnis

Tabelle 1: Wichtigsten Anwendungsfälle für ADAS/AD	25
Tabelle 2: Übersicht des Anwendungsfalls "Navigation mit Echtzeitkarten"	36
Tabelle 3: Bestandteile des V2X	38
Tabelle 4: Übersicht des Anwendungsfalls "V2X am Beispiel Platooning"	47
Tabelle 5: Übersicht des Anwendungsfalls "Smart Parking"	55
Tabelle 6: Abfrageergebnisse	63

1 Einleitung

1.1 Einführung

Automobilhersteller stehen derzeit vor großen Veränderungen. Vor rund 15 Jahren waren assistierende Systeme wie Einparkhilfen, Spurhalteassistenten oder Tempomat in den Fahrzeugen reiner Luxus. Heute sind diese jedoch meistens schon als Teil der Serienausstattung in den meisten Fahrzeugen integriert. Aber nicht nur die in den Fahrzeugen verbaute Technik befindet sich im Wandel, sondern auch das Bewusstsein der Menschen. Wo früher ein lauter Auspuff, große Höchstgeschwindigkeiten und tiefschwarzer Qualm die Kunden gelockt und zufrieden gestellt haben, wächst heute umso mehr das Bewusstsein für die Umwelt und Nachhaltigkeit [1] und auch der Wunsch nach mehr Sicherheit und Komfort. Denn die Anzahl der Fahrzeuge steigt und soll Prognosen zufolge bis 2030 2 Mrd. betragen, wodurch potenziell mehr Verkehrsunfälle, Produktivitätsverlust und Kraftstoffverschwendung entstehen können [2]. Demnach setzen immer mehr Automobilhersteller ihren Fokus auf die Entwicklung nachhaltiger Fahrzeuge, was zum einen über Elektromobilität, zum anderen aber auch über ausgereifte Fahrerassistenzsysteme oder autonomes Fahren erreicht werden kann.

Die Entwicklung solcher Assistenzsysteme ist ein langjähriger Prozess, welcher bereits 1986 begonnen hat und noch immer andauert [3]. Das Problem stellen tatsächlich die Daten dar, welche als Herzstück für die Entwicklung von Assistenzsystemen und dem autonomen Fahren gelten. Die Daten müssen zur Entwicklung spezieller Techniken, wie zum Beispiel der Objekterkennung, Multi-lane detection oder auch der Fahrerüberwachung, über die Sensoren und Kameras der Fahrzeuge, beispielsweise während einer Probefahrt auf den öffentlichen Straßen (real) oder in dafür speziell konzipierten Szenarien (virtuell), aufgezeichnet und anschließend analysiert werden. Hierbei können pro Fahrzeug und pro Fahrzyklus Datenpakete mit einem Umfang von 30 – 60 TB generiert werden, weshalb die Fahrzeuge aktuell mehrfach am Tag

deren Speicher und Daten auf ein Rechenzentrum übertragen müssen, wo diese unter anderem für maschinelles Lernen sowie Deep Learning Verfahren verwendet werden [4]. Hinzu kommt, dass sich auch die Technik in Bezug auf deren Sensoren, Genauigkeit und Auflösung weiterentwickelt, woraus ein höheres Datenaufkommen resultiert [5].

Der Umgang mit diesen Daten stellt einen enormen Aufwand und eine noch größere Herausforderung dar, welcher unter anderem Grund dafür ist, dass die Entwicklung schleppend vorangeht und dementsprechende Technologien zwar immer mehr zur Realität werden, jedoch auch mit großen Kosten verbunden sind. Viele Daten werden derzeit vom Fahrzeug selbst analysiert, woraufhin Entscheidungen getroffen werden. Dies benötigt jedoch sehr viel Rechenpower [6] und dementsprechende Anpassung der Fahrzeuge. Um diese Kosten zu verringern, die Entwicklung zu beschleunigen und die Technik des selbstfahrenden Autos somit den Kunden näher zu bringen, würde es sich deshalb anbieten, einen Teil des Rechenaufwands auf externe Systeme auszulagern.

Um diesen Problemen entgegenzuwirken und effizientere Möglichkeiten zur Analyse, Auswertung und Übertragung der Daten, aber auch der nachhaltigen Speicherung zu schaffen, bietet es sich an, Cloud Computing in die Entwicklung mit einzubinden. Dies würde es möglich machen, gesammelte Daten direkt an die Rechenserver zu übertragen, Tools und Software anderer Cloud-Anbieter für bestimmte Prozesse zu nutzen oder sogar Echtzeitanalysen durchzuführen [3].

1.2 Zielsetzung

Ziel dieser Arbeit ist es, einen prototypischen Kriterienkatalog zu erarbeiten, welcher den Nutzen der Cloud für die ADAS/AD-Entwicklung und speziell einzelnen Anwendungsbereichen wie der Objekterkennung, der Nutzung und Erstellung der HD-Karten oder der Fahrerüberwachung aufzeigt. Dabei sollen eben besagte Anwendungsbereiche näher erläutert und jeweils relevante Cloud-Services und damit verbundene Probleme und Chancen benannt werden. Dieser Katalog soll als Grundlage zukünftiger Entscheidungen im Bezug der ADAS/AD-Entwicklung mit der Cloud dienen.

1.3 Aufbau der Arbeit

Die Arbeit ist in fünf Kapitel aufgeteilt. Im Grundlagenteil wird die Entwicklung und die Unterschiede einzelner Dienstangebote der Cloud miteinander verglichen und analysiert und anschließend die Entwicklung von Fahrerassistenzsystemen (Advanced Driver Assistance Systems, kurz ADAS) und des autonomen Fahrens (Autonomous Driving, kurz AD) erläutert. Auf dieser Grundlage wird als Konzept für verschiedene Anwendungsfälle je ein Vorgehensmodell erarbeitet, welches die Problemdiagnose, Lösungssuche, die Umsetzung und eine abschließende Bewertung beinhaltet. Im Rahmen der prototypischen Umsetzung wird anschließend, auf Grundlage des zuvor erarbeiteten Konzepts, der Nutzen der Cloud für die ADAS/AD-Entwicklung für einen dieser Anwendungsfälle demonstriert. Zum Abschluss der Arbeit findet eine Zusammenfassung der Arbeit und eine Auflistung möglicher zukünftiger Fragestellungen in diesen Zusammenhang statt.

2 Grundlagen

Im Zusammenhang mit den Automobilen beherrschen zwei große Trends die Schlagzeilen. Die Rede ist einerseits von den Elektrofahrzeugen, welche für eine bessere und neutrale CO₂-Bilanz sorgen sollen und andererseits von der Vision des autonomen Fahrens. Letzteres als Symbol für Luxus, der Bequemlichkeit oder auch nur als Stauassistent [7]. Ab Mitte 2017 wurde ein Gesetz zum automatisierten Fahren verabschiedet, welches erstmals den Einsatz von hochautomatisierten Fahrzeugen auf deutschen Straßen gestattet. Vier Jahre später, im Mai 2021, wurde schließlich einem Gesetz zugestimmt, nach dem das vollautomatisierte Fahren (SAE-Level 4) im öffentlichen Straßenverkehr zugelassen wird. Mit diesem Gesetz ist Deutschland weltweit das erste Land, welches fahrerlose Fahrzeuge im Regelbetrieb und gesamten nationalen Geltungsbereich erlaubt [8].

Diese Änderung stellt einen zentralen Baustein für die Entwicklung autonomer Fahrzeuge da. Die Automobilindustrie arbeitet mit Hochdruck daran, den Vorsprung von Tesla zu verringern, um im Bereich des autonomen Fahrens mitzuspielen. Um die Entwicklung voranzutreiben, wird deshalb unter anderem auf den Einsatz von Cloud gesetzt [9].

Im Folgenden wird auf die Eigenschaften des Cloud Computing eingegangen. Hierfür wird zunächst der Hintergrund des Cloud Computing erklärt und im Anschluss die Eigenschaften, Arten und Servicemodelle der Cloud vorgestellt. Der zweite Teil dieses Kapitels behandelt die Hintergründe der Fahrerassistenzsysteme und des autonomen Fahrens. Hierbei wird insbesondere in Punkt 2.2.2 auf die Entwicklung und anschließend auf den aktuellen Stand der Technik eingegangen. Im letzten Punkt wird die Cloud im Kontext der Automobilen vorgestellt, um den Zusammenhang beider Teile zu verdeutlichen.

2.1 Cloud Computing

Die Anforderungen an die IT werden zunehmend größer. Der Bedarf an Speicherkapazitäten, leistungsstärkeren Systemen, die Bewältigung immer größerer Datenströme und eine Systemverfügbarkeit von nahezu 100 % bringen Unternehmen an ihre Grenzen [10]. Das Meistern dieser Anforderungen verlangt neben den hohen Kosten für eine passende Infrastruktur und dessen Verwaltung auch ein entsprechendes Know-how der IT-Abteilung. Dazu kommt das Problem, dass in vielen Fällen eine unregelmäßige Auslastung der Systeme stattfindet. So werden die Server eines Onlineshops beispielsweise zur Weihnachtszeit öfter beansprucht als im Frühjahr. Im Beispiel der Entwicklung für das autonome Fahren würden beispielsweise während Testfahrten größere Datenströme zu bewältigen sein als an Stilltagen. Um mit diesen Aufgaben fertig zu werden, aber gleichzeitig die Kosten und den Aufwand der IT zu minimieren und optimal zu nutzen, setzen immer mehr Unternehmen auf den Einsatz von Cloud-Systemen [11].

Cloud Computing schafft die Möglichkeit, IT-Infrastrukturen über das Internet zu verwenden. Hierbei handelt es sich um Infrastrukturen wie Rechenleistung, Speicher oder sogar ganze Software, welche vom eigenen Unternehmen in das Rechenzentrum des jeweiligen Cloud-Anbieters verlagert wird. Durch Inanspruchnahme dieser Dienste werden die eigenen IT-Dienstleister von der Beschaffung von Hard- und Software befreit und gleichzeitig findet eine Entlastung dieser statt. Durch das flexible Leistungsangebot des Cloud Computing können neue Dienstleistungen wie Speicherressourcen dem jeweiligen Bedarf angepasst und dementsprechend erweitert oder vermindert werden. Somit müssen Unternehmen nicht mehr in den Aufbau oder die Wartung großer Rechenzentren investieren. Die Dienste, Anwendungen und Ressourcen der Cloud-Anbieter stehen den Nutzern dabei über das Internet flexibel und skalierbar zur Verfügung und können somit von jedem Standort aus abgegriffen werden [12].

Die wichtigsten Cloud-Anbieter sind derzeit Amazon Web Service, Microsoft Azure und Google Cloud Platform, welche zusammen Ende 2020 etwa 60 % der weltweiten Cloud-Ausgaben kontrollierten [13]. Der Grund hierfür ist, dass diese Unternehmen vor dem Cloudantritt weltweit führend in ihren jeweiligen Bereichen waren und mit ihrer Innovationskraft und Marktdominanz ideale Grundlagen und Expertise vorwiesen [14].

2.1.1 Begriffe und Hintergrund

Die verschiedenen Technologien des verteilten Rechnens basieren auf der begrenzten Rechenleistung eines einzelnen Computers. Größere Rechenaufgaben, welche mit den Ressourcen eines einzelnen Computers Jahrzehnte zur Berechnung benötigten, können somit in wenigen Tagen oder sogar Stunden ausgeführt werden. Durch dieses Denken entstand die Idee des verteilten Rechnens, die Möglichkeit ungenutzte Ressourcen gewinnbringend einzusetzen und schlussendlich die Cloud [15].

Eine klare Definition des Cloud Computing ist nicht gegeben, da es eine große Variation von Anbietern von Cloud-Lösungen und ihrer Dienste auf dem Markt gibt. In den meisten Fällen bezieht man sich jedoch auf die Definition des National Institute of Standards and Technology (NIST), welche Cloud Computing wie folgt definiert:

„Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model is composed of five essential characteristics, three service models, and four deployment models“ [16]

Es versteht sich damit als Weiterentwicklung des Grid-Computing, wobei auf dessen existierenden Technologien zugegriffen wird. Zudem wird beim Cloud Computing auf einen zentralen Ressourcenbestand eines Anbieters zugegriffen, wohingegen beim Grid-Computing eine dezentrale Zusammenführung mehrerer Ressourcen stattfindet [12]. Damit verspricht Cloud Computing einfacher bedienbar, schneller zugänglich und eine günstigere Alternative entgegen dem Grid-Computing zu sein [15].

2.1.2 Eigenschaften von Cloud Computing

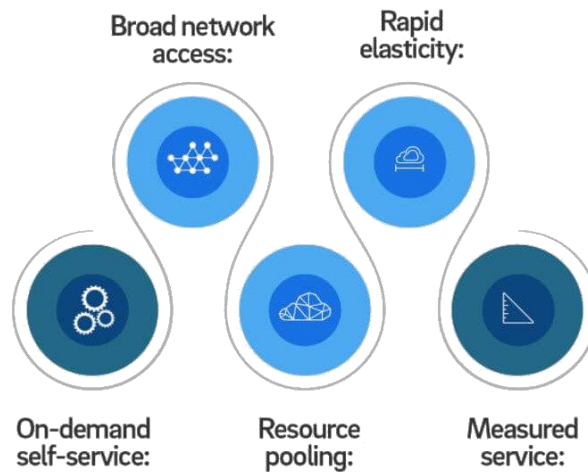


Abbildung 1: Die fünf Charakteristika des Cloud-Computings [17]

Nach dem National Institute of Standards and Technology gibt es beim Cloud Computing fünf Charakteristika, welche die Eigenschaften einer Cloud definieren (Abbildung 1). Zu diesen gehört die Selbstbedienung nach Bedarf (On-demand self-service), ein umfassender Netzwerkzugang (broad network access), die Ressourcenbündelung (resource pooling), eine schnelle Elastizität (rapid elasticity) und die Messbarkeit der Nutzung (measured services) [16].

On-demand self-service: Die Selbstbedienung nach Bedarf bezeichnet die Eigenschaft, dass der Dienstanutzer einer Cloud die angeforderte Kapazität in Eigeninitiative anpassen kann und dabei nicht auf den Cloud-Anbieter angewiesen oder von diesen abhängig ist [18].

Broad network access: Die Cloud-Ressourcen sind (in der Regel) über das Internet erreichbar. Dies kann sowohl über handelsübliche Computer als auch über internetfähige Geräte (wie Tablets, Smartphones oder Laptops) erfolgen. Immer mehr Fahrzeuge werden beispielsweise über SIM-Karten internetfähig. Somit müssen die Daten zukünftig nicht mehr manuell übertragen oder Updates eingespielt werden, sondern können über das Internet stattfinden [19].

Resource pooling: Eine Bündelung der Ressourcen ermöglicht den Nutzern, sich gemeinsam aus einem Ressourcenbestand bedienen zu können. Dieser Vorrat an Ressourcen kann je nach Bedarf auf die Dienstanutzer aufgeteilt werden [16].

Rapid elasticity: Cloud-Dienste können schnell und flexibel an den aktuellen Ressourcenbedarf angepasst werden. Fallen zum Beispiel bei der Entwicklung des autonomen Fahrens während einer Testfahrt unerwartet große Datenmengen an, welche in der Cloud gespeichert werden sollen, so kann der jeweilige Bedarf an Speicher vom Nutzer selbst bzw. über dessen Systeme automatisch vorgenommen werden. Dabei kann im Allgemeinen ein „Scale up“ (Upgrade) oder „Scale down“ (Downgrade) stattfinden [4].

Measured services: Die Nutzung der Ressourcen jedes einzelnen Benutzers wird aufgrund des nutzungsbasierten Abrechnungsmodells (pay-as-you-go) gemessen und kontrolliert. Ebenso wie Strom oder Wasser sollen Cloud-Angebote nutzungsbezogen abgerechnet werden [20] [18].

2.1.3 Cloud Bereitstellungsmodelle

So wie NIST eine Definition zum Thema von Cloud Computing bereitstellt, so unterscheidet die Literatur auch zwischen verschiedene Formen der Cloud. Demnach lassen sich die Cloud-Services in verschiedene Bereitstellungsmodelle unterteilen. Je nach Merkmalsausprägung lassen sich die Cloud-Dienste der Variante einer privatisierten Cloud (Private Cloud), einer öffentlich zugänglichen Cloud (Public Cloud), einer Mischform beider Varianten (Hybrid Cloud) oder als gemeinschaftliche Cloud zuordnen [12]. Eine Übersicht dieser vier Formen wird in Abbildung 2 dargestellt.

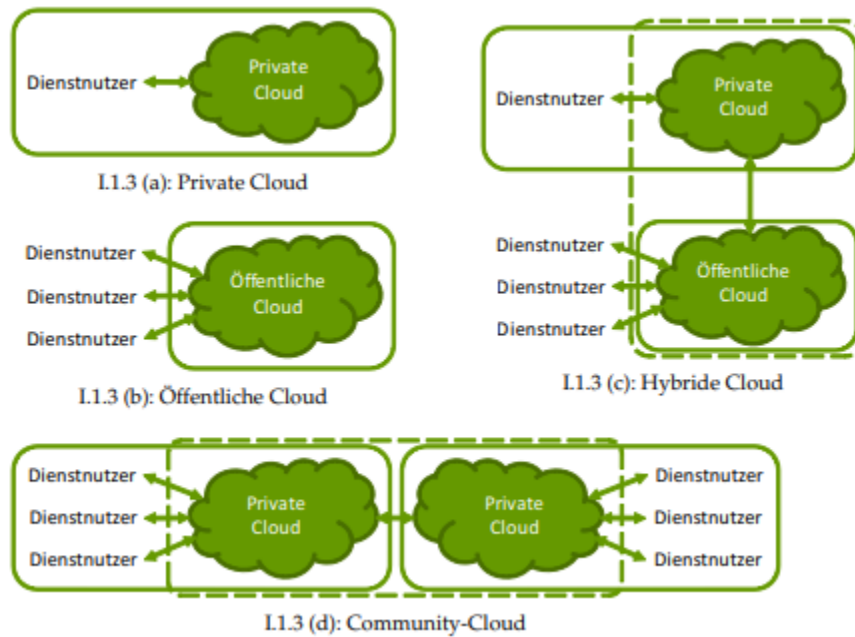


Abbildung 2: Cloud-Betriebsmodelle [21]

Public-Cloud: Bei der Public-Cloud findet die Bereitstellung der Dienste über das öffentliche Internet durch einen Drittanbieter statt. Die genutzten Daten und Dienste liegen in dessen Obhut und stehen für jeden frei zur Verfügung. Aus diesem Grunde stellt die Datensicherheit bei dieser Form das größte Problem dar. Bei diesem Bereitstellungsmodell ist der Anbieter für die Bereitstellung und Wartung der Dienste verantwortlich, wodurch sich Unternehmen auf ihre Aufgaben fokussieren können. Unterschieden wird hierbei zwischen einer exklusiven und einer öffentlichen Cloud. Die exklusive Cloud setzt voraus, dass sich Nutzer und Anbieter kennen und feste Konditionen bezüglich des Preises, der Datensicherheit oder auch des Ausfallschutz ausgehandelt haben. Bei der öffentlichen Cloud hat dieser Austausch nicht stattgefunden, wodurch der Anbieter seinen Service ohne direkten Input des Kunden weiterentwickelt [12] [21] [22].

Private-Cloud: Die Dienste und Services einer Private-Cloud werden exklusiv von einem einzigen Unternehmen genutzt und zugänglich gemacht. Anbieter und Nutzer befinden sich in derselben Organisation, weshalb das Risiko der Datensicherheit und des Datenspeicherorts möglichst gering gehalten wird [21]. Daher eignet sich diese Form der Bereitstellung besonders für Unternehmen, welche sensible Daten pflegen. Die Verantwortung für Kosten und

Verwaltung bleibt in den meisten Fällen jedoch bei der IT-Abteilung des eigenen Unternehmens, wodurch der Aufwand und die Kosten sich ähnlich verhalten wie bei einem herkömmlichen Rechenzentrum [23].

Hybrid-Cloud: Die zuvor vorgestellten Modelle lassen sich zu einer Hybrid-Cloud kombinieren. Dies bringt den Vorteil, dass sensible Unternehmensdaten in der Private-Cloud weiterhin verarbeiten oder gehalten werden können und zeitgleich weniger kritische Daten in einer Public-Cloud betrieben werden [21]. Somit ist es möglich, die Cloud und in erster Linie die Daten dort einzusetzen, wo es wirtschaftlich am sinnvollsten ist und technologische Vorteile bringt. Diese Art der Bereitstellung wird in der Praxis am häufigsten verwendet.

Community-Cloud: Die letzte Art ist ein Cloud-Bereitstellungsmodell, welche auf eine gemeinsame Nutzung der Infrastruktur von mehreren Organisationen setzt. Mehrere Cloud-Anbieter (meist Private-Clouds) schließen sich hierbei zusammen, um für einen ausgewählten Kundenkreis eine gemeinsame Cloud-Dienstleistung zu erbringen [21].

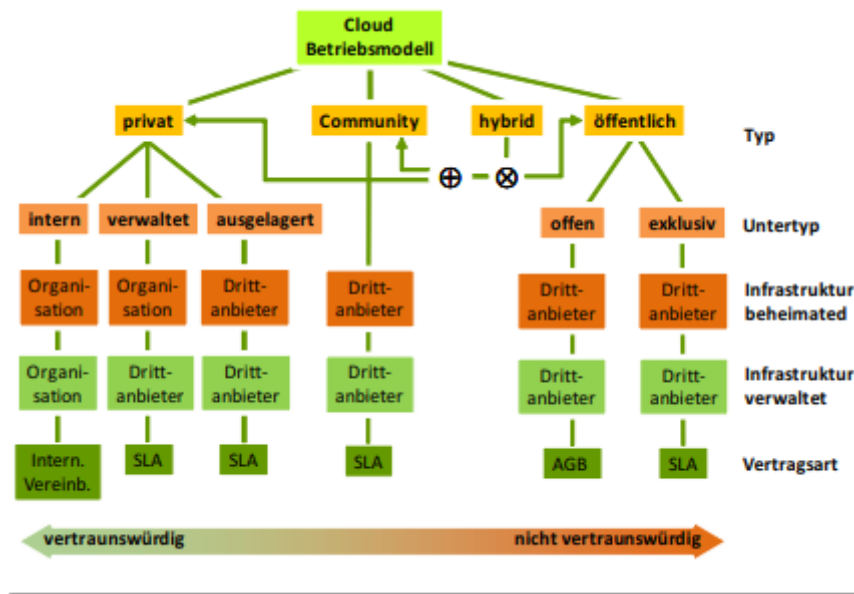


Abbildung 3: Übersicht der Cloud-Bereitstellungsmodelle [21]

Eine Übersicht dieser vier Formen der Bereitstellung ist in Abbildung 3 gegeben. Deutlich zu erkennen ist hierbei, wie sich die verschiedenen Modelle bezüglich der Vertrauenswürdigkeit und deren Art der Bereitstellung/Verfügbarkeit verhalten.

2.1.4 Service Modelle

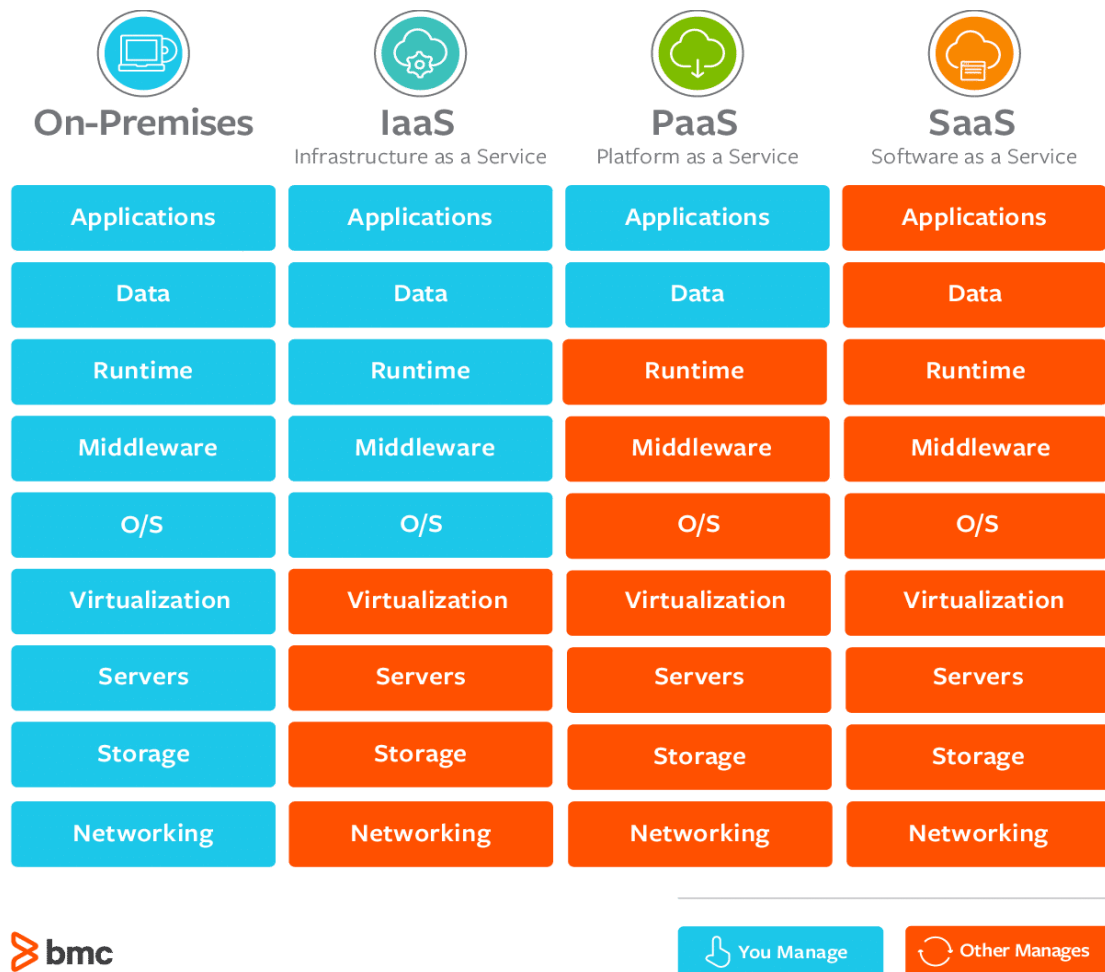


Abbildung 4: Gegenüberstellung der Verantwortlichkeiten für die Servicemodelle der Cloud [24]

Die Bereitstellungen der Services, welche wie zuvor erläutert hauptsächlich über das Public-, Private- oder Hybrid-Cloud-Modell laufen, unterscheidet drei wesentliche Modelle voneinander, welche wie in Abbildung 4 dargestellt aufeinander aufbauen. Durch die Wahl des jeweiligen Modelles wird die Verantwortung zwischen dem Service Providern und dem Kunden bestimmt. Die drei Hauptmodelle sind IaaS (Infrastructure as a Service), PaaS (Platform as a Service) und SaaS (Software as a Service) [25].

Infrastructure as a Service: IaaS ist die Schicht mit dem höchsten Abstraktionsgrad. Es bildet die Grundlage für den Einsatz von Cloud, da hierbei unter anderem eine Bereitstellung von Speicher-, Netzwerk- und Rechenleistung stattfindet [18]. Anwendungen wie Betriebssysteme oder spezielle Software wird dabei nicht vom Anbieter bereitgestellt, wodurch das IT-System vom Dienstleister unabhängig und vom Kunden individuell betrieben wird. Die Abrechnung erfolgt dabei nach dem Pay-per-Use-Modell [26].

Ein typisches Beispiel der Verwendung von IaaS ist die Bereitstellung von Speicherplatz. Die richtige Speicherung und Haltung der Daten spielt insbesondere für die Entwicklung autonomer Fahrsysteme eine gravierende Rolle. Die erhobenen Daten müssen neben der Verwendung z.B. für Analysen auch über die gesamte Lebensdauer eines Fahrzeuges archiviert werden [4].

Platform as a Service: Über der IaaS-Schicht ist die PaaS-Schicht angeordnet. Zusätzlich zur Infrastruktur wird dabei eine Entwicklungs- und Laufzeitumgebung als Service angeboten [25]. In dieser Umgebung wird es den Entwicklern und Benutzern möglich, eigene Software zu entwickeln und auszuführen. Sie erhalten Zugang zu Datenbanken, Middleware, Betriebssysteme und Server ohne die benötigte Konfiguration, Installation und Verwaltung. Konzentriert wird sich ausschließlich auf die Entwicklung und den Betrieb der Anwendung, wodurch die Effizienz gesteigert werden kann [23].

Zielgruppen sind in erster Linie Softwareanbieter, Anwendungsentwickler und Value Added Reseller [12].

Software as a Service: Das Modell „Software as a Service“ besitzt den geringsten Abstraktionsgrad. Es stellt dir Zugang zu einem fertigen Softwareprodukt bereit, welche ebenfalls als Cloud-Service verfügbar sind [25]. Die Verwaltung und den Betrieb übernimmt in dem Fall der Dienstleister, weshalb sich der Anwender vollständig auf die Nutzung der Software konzentrieren kann.

Dies bietet vor allem für Unternehmen einen großen Vorteil, welche ihren Schwerpunkt nicht auf die Entwicklung bestimmter Software legen, sondern z.B. auf die Erhebung und Auswertung von Daten. Somit können Automobilhersteller beispielsweise für ihre erhobenen Daten KI/ML/DL-Tools von anderen Anbietern nutzen und müssen diese nicht selber entwickeln [4].

Durch die Auswahl zwischen den Servicemodellen ist es Unternehmen schlussendlich möglich, ihren Fokus frei zu wählen und sich ggf. auf bestimmte Teile der Entwicklung bzw. des Prozesses zu spezialisieren und eben andere Teile von Drittanbietern zu beziehen.

2.2 Advanced Driver Assistance Systems & Autonomous Driving

In dem nachfolgenden Abschnitt werden die Fahrerassistenzsysteme (ADAS) und das autonome Fahren mit einigen wichtigen Funktionen und Herangehensweisen zur Entwicklung beschrieben.

2.2.1 Überblick

Der Ausbau der Fahrerassistenzsysteme (Advanced Driver Assistance Systems, ADAS) und das darauf aufbauende autonome Fahren (Autonomous Driving, AD) eröffnet eine neue Art der Mobilität. Dabei ist die Vision des fahrerlosen und selbstfahrenden Autos für die Menschen nichts Neues. Schon im frühen 20. Jahrhundert wurde den Menschen unter den Aspekt der Erhöhung der Verkehrssicherheit ein fahrerloses Fahrzeug präsentiert, welches damals jedoch noch per Funk von einem nachfolgenden Fahrzeug gesteuert wurde. Einige Jahre später begann ebenfalls die Entwicklung der ersten Fahrerassistenten, darunter der Tempomat [27].

Diese Entwicklung hat bis heute nicht aufgehört. Fahrzeuge werden in zunehmender Anzahl mit einer Vielzahl von komplexen elektronischen Systemen ausgestattet. Sie sind in der Lage, die Fahrspur zu halten und beim Abkommen der Spur den Fahrer zu warnen bzw. selbst einzugreifen. Sie können vor Kollisionen warnen und schützen, die Geschwindigkeit der Umgebung und Verkehrslage anpassen und besitzen viele weitere Funktionen. Studien zufolge werden aktuell bis zu 90 % der Unfälle auf den Straßen aufgrund von menschlichem Versagen verursacht und nur etwa 2 % durch Geräteausfälle [28].

Somit wird ADAS als Forschungsgebiet im Bereich „Intelligent Transporting Systems“ immer wichtiger. Man will damit für mehr Sicherheit im Verkehr und einen geringeren Treibstoffverbrauch sorgen [29]. Doch der Weg von einzelnen Assistenzsystemen hin zum AD ist schwer, da die Anforderungen dabei sehr hoch sind. Demzufolge sagt man, dass die automatisierten Systeme mindestens doppelt so gut sein müssen wie ein Mensch [30].

Doch um das zu erreichen, werden viele Daten benötigt, welche dementsprechend optimal verwendet und analysiert werden müssen.

Stufen der Autonomie:

Der Umfang dieser Daten, die Möglichkeit des Fahrzeuges, die Aufgaben des Fahrers bei Bedarf zu übernehmen und zukünftige Interaktionen zwischen Mensch und Maschine auf der Straße geschieht in unterschiedlichen Entwicklungsschritten. Diese Entwicklungsschritte lassen sich in fünf Stufen/Level unterscheiden.



Abbildung 5: Die fünf Stufen der Autonomie [31]

Level 1: Assistiertes Fahren

- Fahrer muss den Verkehr ständig im Blick behalten
- Fahrer haftet für Verkehrsverstöße und Schäden
- Einzelne Assistenzsysteme unterstützen bei bestimmten Fahraufgaben

Das assistierte Fahren ist schon heute in vielen Fahrzeugen Realität. Hierbei wird der Fahrer von den Systemen unterstützt, welche jedoch nicht die Steuerung übernehmen. Beispiele in der Praxis sind dabei das Antiblockiersystem, ein Tempomat oder auch ein Spurhalteassistent. Die Komplexität der Entwicklung und benötigten Daten ist dabei überschaubar, da die einzelnen Systeme nicht voneinander abhängig sind und den Fahrer im Regelfall nur informieren [32] [33]. Für die Umsetzung sind bei diesem Level in der Regel ein bis zwei Sensoren ausreichend [34].

Level 2: Teilautomatisiertes Fahren

- Fahrer muss den Verkehr ständig im Blick behalten
- Fahrer haftet für Verkehrsverstöße und Schäden
- Systeme vom Fahrzeug können das Steuer übernehmen

Im Gegensatz zum Level 1, greift das Fahrzeug beim teilautomatisierten Fahren unter bestimmten Bedingungen ins Verkehrsgeschehen ein. Zum Beispiel kann ein Fahrzeug der Stufe 2 auf der Autobahn die Spur halten und gleichzeitig bremsen oder beschleunigen. Damit dies funktioniert, werden verschiedene Einzelsysteme miteinander gekoppelt [35] [33].

Level 3: Hochautomatisiertes Fahren

- Fahrer kann sich in bestimmten Situationen länger vom Fahrgeschehen abwenden
- Das Fahrzeug kann in vorgegebenen Anwendungsfällen selbstständig fahren
- Fahrer muss nach Aufforderung des Systems das Steuer übernehmen, andernfalls haftet er selbst.

Ab Level 3 gewinnt der Fahrer zunehmend mehr Freiheit. Das Fahrzeug kann in bestimmten Situationen, wie beispielsweise auf der Autobahn oder im Stau die Fahraufgaben vollständig übernehmen [36]. Der Fahrer muss aber innerhalb von Sekunden in der Lage sein, das Steuer wieder zu übernehmen. Diese Aufforderung kann z.B. kommen, wenn das System ein Fehler erkennt, die Autobahn verlässt oder unerwartet auf eine Baustelle trifft [32] [33].

Level 4: Vollautomatisiertes Fahren

- Fahrer kann die Steuerung über längere Distanzen und in verschiedenen Verkehrssituationen abgeben und wird zum Passagier
- Als Passagier kann der Fahrer sich vollständig vom Verkehrsgeschehen abwenden
- Fahrer haften während der vollautomatisierten Fahrt nicht für Verkehrsverstöße oder Schäden

Level 4 ist die Vorstufe zum autonomen Fahren. Dabei sind die technischen Systeme vom Fahrzeug in der Lage, selbstständig zu navigieren und sogar hochkomplexe urbane Verkehrssituationen, wie zum Beispiel auftretende Baustellen, zu meistern [33].

Das Fahrzeug ist dazu fähig, auf die Autobahn aufzufahren, sich in den Verkehr einzuordnen, der Spur zu folgen, zu blinken, zu überholen, bei Bedarf zu bremsen und die Autobahn schließlich zu verlassen. Erkennt das System jedoch Situationen, bei denen es an seine Grenzen stößt, wird der Fahrer zur Übernahme aufgefordert oder das Fahrzeug selbstständig in einen sicheren Zustand überführt.

Die meisten Entwicklungsabteilungen der großen Autokonzerne konzentrieren sich momentan auf genau diesen Teil [32]. Dabei sind jedoch im Gegensatz zur Stufe 1 bis zu 64 Sensoren nötig, die das Fahrumfeld überwachen, analysieren und nächste Aktionen berechnen. Dabei fallen riesige Mengen an Daten an, welche schnell und zuverlässig verarbeitet werden müssen [34].

Level 5: Autonomes Fahren

- Es gibt nur noch Passagiere und keine Fahrer
- Fahrzeuge können ohne Insassen fahren
- Passagiere haften für keine Verkehrsverstöße oder Schäden

Die fünfte und letzte Stufe lässt selbst das Gaspedal und Lenkrad aus den Fahrzeugen verschwinden. Das Fahrzeug übernimmt jetzt alle Fahraufgaben und die Personen im Wagen werden zu Passagieren, welche weder eine Fahrtüchtigkeit noch eine Fahrerlaubnis benötigen [32] [33].

Die Realisierung ist dementsprechend sehr komplex und die Anforderungen sehr hoch. Nicht ohne Grund arbeitet die Autonomieindustrie schon Jahrzehnte an dessen Umsetzung. Doch durch den rasanten Fortschritt und der Leistungsstärke der Technik und seiner Systeme wird von vielen Automobilherstellern angekündigt, Fahrzeuge der Stufe vier und fünf in den nächsten zehn Jahren auf den Markt zu bringen. Die Schlüsseltechnologie soll dabei eine High-Definition-Karte bilden, welche eine präzise Lokalisierung des Fahrzeuges und eine Erkennung und Reaktion auf Ereignisse und Hindernisse ermöglichen soll. Neben dem Kennen der Straße und Hindernisse ist für das vollautomatisierte und autonome Fahren zusätzlich eine schnelle Reaktionszeit wichtig. Hindernisse müssen identifiziert und ggf. ausgewichen werden [37].

Um diese Funktionalität zu erreichen und einen technisch erforderlichen Reifegrad sicherzustellen, ist eine hohe Anzahl an Testfällen nötig. Diese wird durch Wachenfeld und Winner für eine Absicherung eines Versionsstandes des Systems auf ungefähr 6,62 Milliarden Testkilometer geschätzt [38]. Zusätzlich muss diese Distanz, für eine erneute Absicherung, bei jeder Modifikation erneut gefahren werden. So eine große Zahl an Testkilometern lassen sich im Realverkehr nur sehr umständlich und mit hohem Aufwand bewerkstelligen. Daher werden als Alternative häufig sogenannte X-in-the-Loop Verfahren angewendet. Das X steht dabei charakteristisch als Sammlung der Systembestandteile, wie zum Beispiel dem Driver, der Software oder der Hardware. Letztere beiden werden im Folgendem, im Zusammenhang mit der Cloud, vorgestellt [39].

Software-in-the-Loop

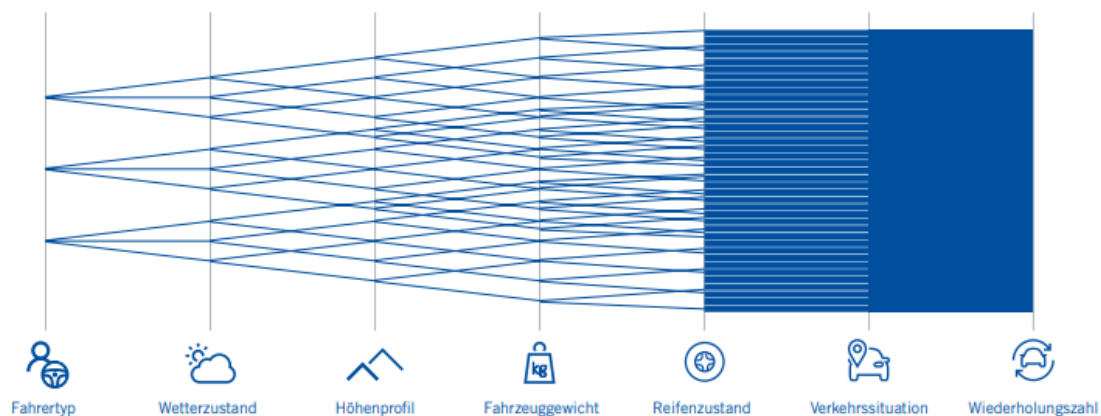


Abbildung 6: Der Zuwachs an zu berücksichtigten Testfällen durch die Kombination von Parameter [40]

Das Software-in-the-Loop (SIL) Verfahren simuliert alle Softwareelemente, welche zum autonomen Fahren oder für einzelne Fahrerassistenzsysteme benötigt werden. Dazu zählt unter anderem das Fahrverhalten, das Fahrzeug, die Fahrdynamik, die Wahrnehmung des Fahrzeuges zu seiner Umgebung und die Szenerie [39]. Millionen von Testkilometern können dadurch virtuell und für die verschiedensten Szenarien, Umgebungsbedingungen und insbesondere auch Grenzfälle und kritische Situationen gefahren werden [41]. Durch die Verschiebung der Testkilometer von der realen Straße auf virtuelle Simulationen können Ressourcen wie Tank und Fahrzeuge reduziert und eingespart werden [42]. Gefährliche Fahrmanöver, physikalische

Grenzen der Fahrdynamik und auch der Ausfall einzelner Systeme kann getestet werden, was in dieser Form auf realen Straßen nur schwer vorstellbar ist [40]. Die Anzahl der Testfälle und damit auch die nötige Rechenleistung nimmt jedoch mit der Zeit zu, was in Abbildung 6 deutlich zu erkennen ist. Die Folge daraus ist, dass die lokal installierten Testsysteme die SIL-Simulationen nicht mehr effizient durchführen können.

Die Lösung dabei ist, benötigte Rechenleistung in die Cloud zu verlagern. Testumfänge würden sich dadurch enorm ausweiten lassen, da nicht mehr auf Rechnern mit begrenzten Ressourcen gerechnet wird, sondern auf skalierbaren Cloud-Infrastrukturen [41]. Einzelne Funktionen können effizient, einzeln und anschließend in Kombination getestet werden. Sofern diese positiv getestet wurden und den erforderlichen Reifegrad erfüllen, können sie auf die Fahrzeuge übertragen werden.

Hardware-in-the-Loop

Bei dem Hardware-in-the-Loop (HIL) -Verfahren werden einzelne Steuergeräte des Fahrzeuges auf ihre Funktionsweise getestet. Diese umfassen neben der Lenkung, Bremse und elektrischen Antrieb auch die Fahrdynamik sowie sämtliche Sensoren [41]. Dabei wird die zu testende Hardware jedoch nicht direkt an dem realen System getestet, sondern an einen dafür vorgesehenen Teststand eingebunden. Der Hardware werden dabei reale Bedingungen vorgespielt, so dass aus seiner Sicht kein Unterschied zwischen Teststand und späterer Regelstrecke besteht [43]. Die Hardware-Elemente werden meistens erst im späteren Prozess der Entwicklung den Testfällen hinzugefügt. Begonnen wird in den meisten Fällen mit dem Testen der Software. Anschließend kann diese in Zusammenarbeit mit der Hardware getestet werden. Dies soll verhindern, dass Hardwareteile durch ein zu spätes Testen nachträglich geändert werden müssen, was von großen Kosten und Verzögerung begleitet wäre.

Es wird deutlich, dass ein schnelles und sicheres Testen ein gravierender Prozess zur Entwicklung von ADAS und AD ist. Manche Anwendungen müssen mehr und andere weniger getestet werden. Klar ist jedoch, dass ausgiebige Tests nötig sind, um alle Fahrscenarien zu validieren. Mit SIL- und HIL-Systemen wird es, wie eben aufgezeigt, möglich, aufwendige und teure Tests von der Straße in die virtuelle Umgebung zu holen [44].

2.2.2 Entwicklung

Die Entwicklung der autonomen Fahrzeuge hat in den letzten Jahrzehnten eine immer wichtiger werdende Rolle in der Automobilindustrie und Gesellschaft eingenommen [3]. Denn die Anzahl der Fahrzeuge auf den Straßen steigt. Wo sie 2002 noch bei etwa 800 Mio. Fahrzeugen auf den Straßen lag, sollen Prognosen zufolge 2030 schon 2 Mrd. Fahrzeuge weltweit zugelassen sein [45]. Dies kann sowohl für Probleme in der Auslastung des Verkehrsnetzes als auch in der Sicherheit mit sich bringen.

Dank der neusten Technik und neuen Verfahren hat die Zahl der Fahrzeuge mit ADAS stark zugenommen und auch Fahrzeuge der Stufe 4 und 5 können schon in naher Zukunft Realität werden. Die Rede dabei ist von Deep-Learning Verfahren, Generierung von 3D-Welten und dafür notwendige Sensoren. Die Fahrzeuge werden, wie in 2.2.1 beschrieben, mit immer mehr Sensoren ausgestattet. Diese Sensoren, Kameras, etc. ermöglichen einen 360° Blickwinkel in und um das Fahrzeug [46]. Dadurch entstehen bei jeder Testfahrt riesige Datenmengen, welche es zu verarbeiten gilt. Dies geschieht durch den Einsatz von Deep-Learning-Verfahren. Jene sind in der Lage, die riesigen Datenmengen zu verarbeiten und in Muster und Modelle abzuleiten. Sie werden dabei auf ausgewählte Szenarien trainiert. Anhand dieser Muster und dem Training ist man dann in der Lage, Entscheidungen für bestimmte Situationen im Straßenverkehr zu treffen, welche vorher noch vom Fahrer getroffen wurden. Die Befehle werden dementsprechend an zentrale Steuereinheiten und das Fahrsystem übertragen. Sie bildet die Kern-technologie beim AD und auch einzelnen ADAS [5].

Ein weiterer Aspekt ist die Generierung von 3D-Welten bzw. High-Definition-Karten, welche eine reale Abbildung von Straßen und Hindernissen bildet. Diese hilft dem Fahrzeug und seinen Steuersystemen, sich beispielsweise auf unebene Straßen, unübersichtliche Verkehrslagen oder scharfe Kurven einzustellen.

Als Unterstützung dieser Systeme kann wieder einmal eine Cloud-Plattform eingesetzt werden. Diese kann sowohl als Datenspeicher, für die Simulation, als Rechenstütze für Deep-Learning-Verfahren oder auch als gemeinsame Nutzung der 3D-Welt genutzt werden. Somit können beispielsweise mehrere Fahrzeuge auf ein und dieselbe virtuelle 3D-Welt zugreifen. Sie würden über die Cloud neue Informationen über kommende Verkehrsinfrastruktur einbeziehen und

können gleichzeitig neue Updates, wie zum Beispiel neu entstandene Schlaglöcher oder auch aufgebaute Baustellen in die Cloud für nachfolgende Fahrzeuge bereitstellen.

Auf der anderen Seite können durch die Rechenleistung der Cloud die Deep-Learning oder ähnliche, auf künstliche Intelligenz zurückgreifende Verfahren effizienter eingesetzt werden und durch intensiveres Training auch schneller zu Ergebnissen kommen [47].

2.2.3 Stand der Technik

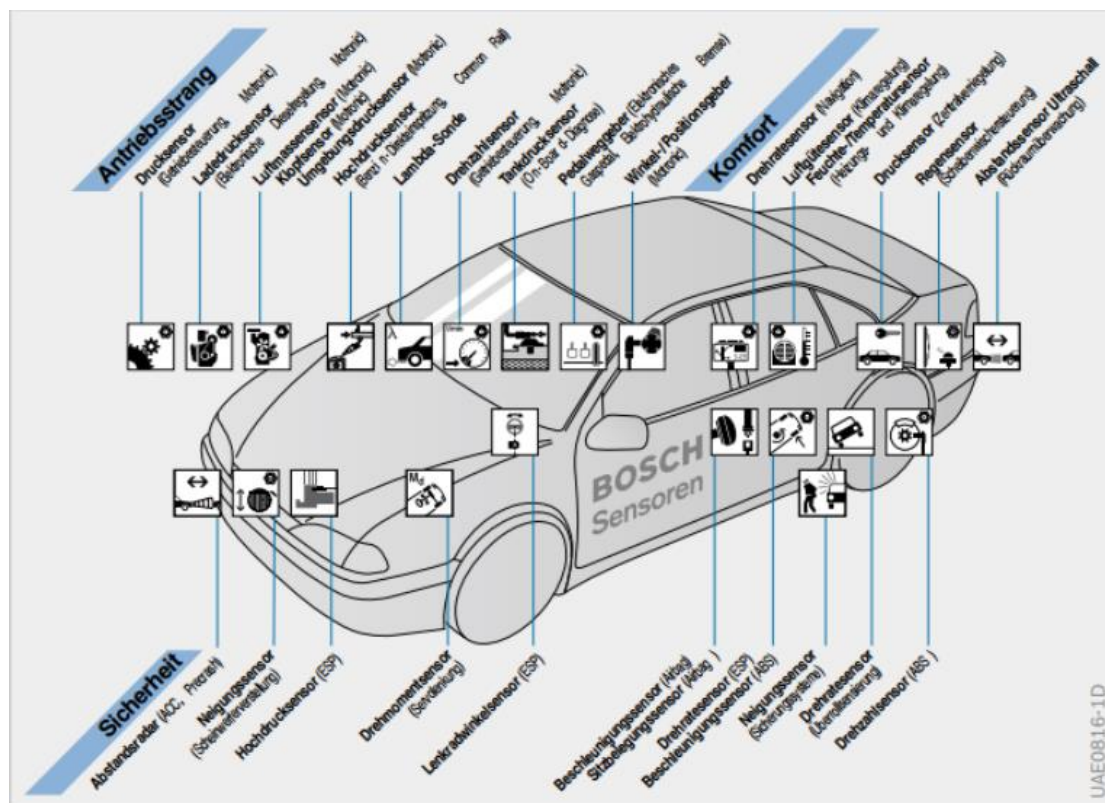


Abbildung 7: Vielfalt der Technik und Sensoren in neuen Fahrzeugen [48]

Vom aktuellen technischen Stand ausgegangen, ist autonomes Fahren schon heute möglich [49]. Moderne Fahrzeuge sind mit einer Reihe von Technik und Sensoren ausgestattet (Abbildung 7). Sie verfügen über Erfassungs-, Netzwerk- und Datenverarbeitungsfunktionen [50]. Dies sind für das autonome Fahren wichtige Bausteine, welche relevant sind für Systeme wie:

- Cruise Control
- Spurstabilitätssysteme
- Parksysteme
- Antischlupfregelung (ASR)

Diese können darüber hinaus auch durch Kooperation zu „neuen“ Systemen vereint werden. Somit kann beispielsweise das ABS und die Lenkradsteuerung zu einem intelligenten Bremssystem kombiniert werden [46]. Den Systemen ist es auch schon jetzt möglich, Schilder zu erkennen, vor einer Haltelinie zu halten und Kreuzungen zu beachten und ggf. „rechts vor links“ zu gewähren. All dies ist durch den erfolgreichen Einsatz von KI möglich [49].

Doch die Technik ist noch nicht ausgereift genug und die Systeme stoßen noch oft an ihre Grenzen. So zeigen Beispiele von Teslas Autopiloten, dass dieser Innerhalb von Städten noch oft an seine Grenzen stößt und der Eingriff des Fahrers unabdingbar ist. Zudem bleibt immer ein Restrisiko bezüglich der Genauigkeit der Systeme und wie diese im Ernstfall zu reagieren haben. Somit kann beispielsweise eine Zeitung, welche über die Straße geweht wird mit einem kleinen Kind verwechselt werden, wodurch das Fahrzeug eine Vollbremsung einlegen würde und nachfolgende Fahrzeuge in Gefahr laufen, hinten aufzufahren. Andersherum kann jedoch auch ein Kleinkind mit einer Zeitung verwechselt werden. Das Fahrzeug würde in dem Fall normal weiterfahren und könnte eine Gefahr für das Kind darstellen [46]. Dies ist nur ein Beispiel davon, an welche Grenzen Fahrsysteme Stoßen.

Um dieses Risiko zu minimieren, muss die Technik weiter ausgereift sein. Die KI muss weiter trainiert werden und die Systeme zuverlässiger arbeiten. Mit Zuwachs der Systeme und Daten steigt andererseits auch der Rechenbedarf [31].

Die meisten Automobilhersteller sind nach aktuellem Stand noch nicht so weit, autonome Fahrzeuge auf den Markt zu bringen. Der Grund dafür ist oftmals der Mangel an Testkilometern. Google selbst besitzt eine zu kleine Flotte, um genügend Testkilometer zu sammeln und auch Geodatendienste wie HERE, welcher ursprünglich von Nokia entwickelt wurde, erreicht mit dessen Flotte von 400 Fahrzeugen und 3 Mio. kartierten Kilometern noch keine ausreichenden Ziele [37]. Dennoch planen Unternehmen wie VW 2025 ihre ersten autonomen Fahrzeuge auf die Straße zu bringen [51].

2.3 Cloud Computing im Kontext der automobilen Assistenzsysteme

Wie aus dem bisherigen Kapitel hervorgeht, spielen Rechenleistung, Daten und deren Umgang eine wichtige Rolle in der Entwicklung und Ausführung von ADAS und AD. Zudem wird es nicht mehr ausreichen, das Fahrsystem allein auf das Sichtfeld seiner Sensoren zu beschränken. Es wird vielmehr auch Informationen von außerhalb wie zum Beispiel der Verkehrsinfrastruktur oder von in der Nähe befindenden Fahrzeuge benötigen [52].

Aus diesem Grund setzen sich immer mehr Automobilhersteller mit Cloud-Herstellern zusammen. So zum Beispiel Microsoft und VW, welche 2020 eine Partnerschaft und die Volkswagen Automotive Cloud bildeten oder die Zusammenarbeit zwischen Amazon mit seinem Amazon Web Services und Ford [9]. Eine solche Zusammenarbeit am Beispiel von VW und Microsoft für die Entwicklung des autonomen Fahrens ist in Abbildung 8 visualisiert. Dabei werden Sensor- und Systemdaten aus dem Echtzeitverkehr in die Cloud übertragen, woraufhin Tools und Methoden für die Funktionen vom AD bereitgestellt und entwickelt werden.

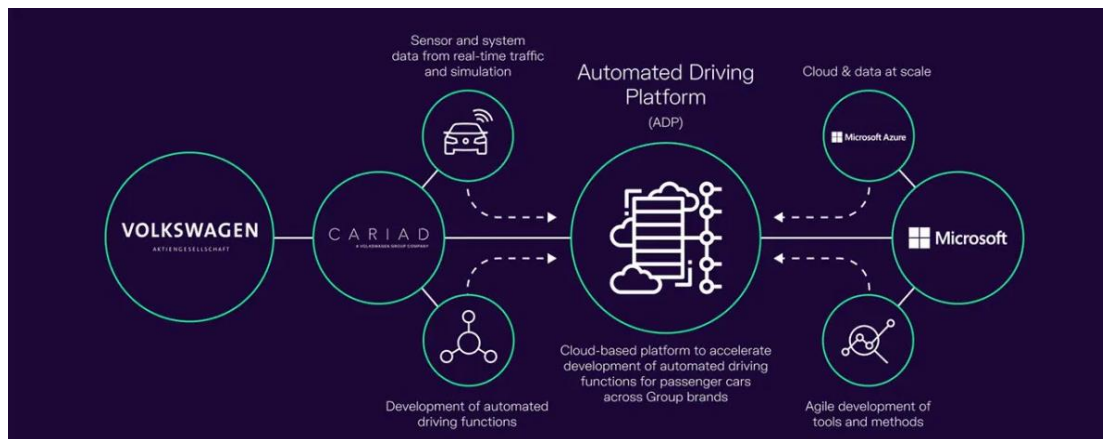


Abbildung 8: Volkswagen Partnerschaft mit Microsoft für die Entwicklung des autonomen Fahrens [53]

Eine wichtige Frage, welche es in diesen Kontext zu beantworten gilt, ist, ob durch den Einsatz von Cloud auch die gewünschten Ziele erreicht werden können. Auch wenn die Cloud ein hohes Maß an Flexibilität und Rechenleistung bietet und die Berechnungen dadurch hoch präzise werden, wird die Cloud aktuell vorrangig für Multimedia- und Navigationssysteme verwendet

[28]. Viele ADAS und AD-Anwendungen verlangen jedoch eine Echtzeitverarbeitung. Die Latenz spielt dabei also eine große Rolle. Zudem ist der Cloud-Markt und seine Auswahl riesig und wächst stetig weiter [54]. Daher wird es zunehmend schwieriger, den „richtigen“ Cloud-Anbieter/-Service zu finden.

3 Konzeption

Nachdem im letzten Kapitel die Grundlagen des Cloud Computing und des ADAS/AD beschrieben und bekannte Probleme benannt wurden, soll im Folgenden unter Einbeziehung dieser Erkenntnisse die verbesserte Entwicklung einzelner Use-Cases von Fahrerassistenzsystemen und des autonomen Fahrens durch die Cloud konzipiert werden. Dazu werden im ersten Schritt die Anforderungen und Probleme der jeweiligen Systeme vorgestellt. Anschließend werden in der Lösungssuche die dafür notwendigen Maßnahmen zur Beseitigung der Probleme erläutert. Dabei wird aufgezeigt, für welche Szenarien sich der Einsatz einer bestimmten Cloud lohnt und für welche nicht.

3.1 Eigenschaften des Konzepts

Die Möglichkeiten, um die Entwicklung der Fahrerassistenzsysteme und des autonomen Fahrens voranzutreiben, sind ebenso vielfältig wie das Angebot an Cloud und anderen Services, welche man dafür in Betracht ziehen kann. Ein Grund dafür ist die Vielzahl an Systemen, welche in aktuellen Autos verbaut sind. Sie reichen von einfachen Assistenten, welche den Fahrer über den Verbraucher- oder Systemstatus des Fahrzeuges informieren bis hin zu hochkomplexen Systemen, welche die Position des Fahrzeuges bis auf wenige Zentimeter bestimmen oder das Ebenbild der Straße abbilden. Gerade deshalb ist es wichtig, einen Überblick über den aktuellen Stand der Dinge zu erlangen, um das optimale Angebot auswählen zu können.

Im Folgenden werden drei ausgewählte Use-Cases (dargestellt in Tabelle 1) für ADAS und AD vorgestellt. Dabei sind Anwendungen, welche wichtig für die Sicherheit des Fahrers, die Sicherheit anderer Verkehrsteilnehmer und für den Betrieb sind, weiter oben eingeordnet, wohingegen Anwendungen, welche eher dem Luxus und der Bequemlichkeit dienen, weiter unten eingeordnet sind. Unterschieden wird zudem zwischen drei Kategorien:

- **Fahrerübernahme:** Die Übernahme der Fahrfunktionen erfolgen durch das System. Dabei kann dieses entweder den Fahrer unterstützen oder komplett von seinen Fahraufgaben befreien.
- **Notfallassistent:** Der Notfallassistent greift nur in besonderen Situationen ein, bei denen die Sicherheit des Fahrers oder seines Umfeldes gefährdet anzusehen sind.
- **Fahrerunterstützung:** Systeme, welchen den Fahrer lediglich über bestimmte Situationen in Kenntnis setzt. Dabei kann es sich beispielsweise um vorausliegende Gefahrensituationen handeln.

Tabelle 1: Wichtigsten Anwendungsfälle für ADAS/AD

Use-Case	Kurzbeschreibung	Kategorie
Navigation	Navigiert das Auto von Start zum Zielpunkt. Dabei werden auch Straßengegebenheiten aufgezeichnet, Geschwindigkeitsbegrenzungen eingehalten und die Position des Fahrzeuges lokalisiert.	Fahrerübernahme, Notfallassistent
V2X	Ermöglicht eine Kommunikation zwischen Fahrzeug und der Umgebung zur rechtzeitigen Warnung und Kenntnisnahme anderer Verkehrsteilnehmer.	Fahrerunterstützung
Smart Parking	Teilt Informationen über freie Parkplätze an umliegende Fahrzeuge mit.	Fahrerübernahme, Fahrerunterstützung

Für diese drei Anwendungsfälle soll je ein Konzept erarbeitet werden. Um dies zu erreichen, werden zunächst die Probleme der Systeme kritisch untersucht und anschließend eine Lösungssuche vorgenommen. Abbildung 9 bietet einen Überblick über das weitere Vorgehen:

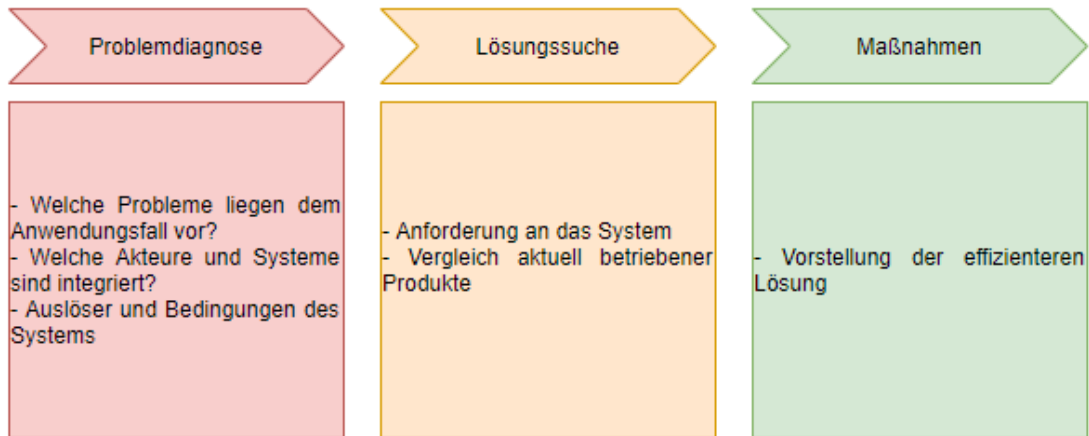


Abbildung 9: Vorgehen des Konzepts

3.2 Konzept zur Optimierung der ADAS/AD-Entwicklung

Im Folgenden werden die drei zuvor beschriebenen Anwendungsfälle im Hinblick auf eine Verbesserung ihrer Entwicklung durch die Cloud konzipiert.

3.2.1 Navigation mit Echtzeitkarten

Wichtige Fragen des autonomen Fahrens für das Fahrzeug sind, wo sich dieses momentan befindet, wo dessen Ziel ist und wie es schlussendlich am schnellsten dorthin kommt. Denn die Fahrt beginnt mit dem Punkt, an dem das Fahrzeug startet und endet, sobald dieses sicher abgestellt wurde. Deshalb ist es für ein Fahrzeug und seine Systeme wichtig, zu jedem Zeitpunkt seine eigene Position, die dortigen Gegebenheiten und den jeweiligen Weg zu seinem Ziel zu kennen.

Damit die Navigation des Fahrzeuges gewährleistet werden kann, muss im Optimalfall ein exaktes Abbild der Verkehrsinfrastruktur vorliegen, aus welchem sich Straßengegebenheiten,

Verkehrsschilder, Ampeln, gefährliche Situationen, aber auch Gebäude, andere Bauten und Bäume ableiten lassen. Die Systeme des Fahrzeuges könnten dann vor und während jeder Fahrt auf eben diese Informationen zurückgreifen und eine autonome Fahrt ermöglichen. Dabei würde dieses dann eigenständig die Route planen, Überhol- oder Spurwechselmanöver durchführen und dabei auf Verkehrsregeln und sein Umfeld achten. Signifikante Punkte wie Gebäude, Bäume oder Straßenführungen können dabei als Referenzpunkte in den Karten markiert werden, um mit dessen und der Hilfe der Sensoren eine Lokalisierung des Fahrzeuges durchführen zu können. Dies kann mithilfe einer digitalen 3D-Karte erreicht werden, welche sich mithilfe der eingebauten Sensoren und Kameras der Fahrzeuge stetig weiterentwickelt und erneuert wird. Inwiefern der Einsatz einer Cloud dabei nützlich ist, wird im Folgenden konzipiert.

Problemdiagnose

Die Einführung des autonomen Fahrens wird ein schwerer Schritt. Die ersten autonom befahrbaren Gebiete, neben den Autobahnen, werden höchstwahrscheinlich extra dafür ausgewählte und abgegrenzte Bereiche sein, welche über eine hochauflösende und hochgenaue Karte verfügen, die alle Umgebungsbedingungen und die aktuelle Verkehrslage vollständig erfasst. Denn es wird nötig sein, die Position des Fahrzeuges zu jedem Zeitpunkt exakt ermitteln zu können und neben den Sensoren und Kameras auch darüber hinaus über die Verkehrsbedingungen informiert zu sein. Karten, welche zuvor hauptsächlich vom Menschen genutzt wurde, um Orte zu finden, Sehenswürdigkeiten zu betrachten oder auch nur seine eigene Straße anzuschauen, werden zukünftig von Maschinen verwendet. Dazu kommt, dass aktuelle Techniken wie das Global Positioning System (GPS) künftig nicht mehr ausreichen werden, um bei der Navigation, Richtungs-, Geschwindigkeits- oder Positionsbestimmung helfen zu können. Der Grund dafür ist, dass diese lediglich eine grobe Position mit einer Genauigkeit von mehreren Metern liefern. Bei der Navigation von Fahrzeugen ist jedoch eine Genauigkeit auf Zentimeterebene erforderlich, damit das Fahrzeug in Kurven nicht ausbricht oder andere Verkehrsteilnehmer gefährdet [55].

Die aktuelle Technik trifft jedoch auch an anderen Orten auf Hindernisse. Die größten Herausforderungen in der Erstellung von High-Definition Maps kurz aufgelistet sind:

- **Datenerhebung**

- **Datenverarbeitung**
- **Datenübertragung**
- **Latenzzeit**

Datenerhebung: Um die, für eine HD-Karte benötigten Daten, zu sammeln, müssen hochausgerüstete Fahrzeuge mehrere Millionen Kilometer Strecke zurücklegen. Dabei werden durch Sensoren und Kameras mehrere Terabyte an Daten täglich erfasst. Die Kerntechnologie bildet dabei die kostspielige Lidar-Technik, welche Laserstrahlen abwirft, um damit die Entfernung, Proportion oder Geschwindigkeit von Objekten zu erfassen [56]. Hierbei entstehen pro Sekunde mehr als zwei Millionen Messwerte, welche es auszuwerten und zu analysieren gilt.

Datenverarbeitung: Nachdem die Daten erfasst wurden, müssen diese anschließend interpretiert werden. Dies kann im Fahrzeug oder außerhalb des Fahrzeuges stattfinden. Die Verarbeitung von einem Terabyte an Daten ist jedoch sehr rechenintensiv und kann deshalb unter normalen Umständen und auch mit hoher Rechenleistung mehrere Tage in Anspruch nehmen [37]. Darüber hinaus erfordert eine höhere Rechenleistung mit dementsprechender GPU und CPU auch einen immens höheren Stromverbrauch. Der Verbrauch moderner GPUs liegt beispielsweise bei 200-300 Watt [57]. Hierbei stellt sich dann die Frage, wie viel Rechenleistung in ein Fahrzeug integriert werden kann, ohne dass sich dieses dadurch selbst tötet. Denn auch Systeme wie Wischeranlagen, Sitzheizung und andere Assistenzsysteme verbrauchen erhebliche Mengen an Strom. Als Vergleich kann man sagen, dass der allgemeine Verbrauch von 100 Watt etwa einem Spritverbrauch von 0,1 Liter auf 100 km entspricht [58].

Ein weiteres Problem stellt die Art der Verarbeitung dar. Es wird dementsprechende Software und Programme benötigt, welche die erhobenen Daten aufbereiten und in Form einer High-Definition-Karte darstellen und umsetzen können.

Datenübertragung: Nachdem oder bevor die Daten verarbeitet bzw. erhoben wurden, müssen diese auf jeweilige (Cloud-)Server übertragen oder von (Cloud-)Servern zu den entsprechenden Fahrzeugen transferiert werden, damit dessen Systeme die aktuelle Karte erhalten. Üblicherweise geschieht dies drahtlos. Die aktuelle 4G-Technologie ermöglicht dabei jedoch nur Übertragungen von bis zu 100-150 MBit/s [59], wobei die Anforderungen der Übertragung bei mehreren Gigabytes liegen.

Latenzzeit: Sofern eine Echtzeitausführung benötigt wird, um Echtzeitinformationen der Verkehrsumgebung zu erhalten wie beispielsweise Stauenden, Unfälle, usw., werden geringe Latenzen von bis zu 10ms erforderlich. Das Problem dabei ist jedoch, dass diese je nach Ort und Tageszeit sehr unterschiedlich ausfallen können. Wie gravierend sich Netzwerküberlastungen und Latenzprobleme auf die Steuerung von Fahrzeugen auswirken können, wird in [60] deutlich. Das Steuern eines Kleinfahrzeuges mit einer Latenz von 150ms von der Cloud aus verursacht eine Kursabweichung von 0,096m auf 40% der Strecke. Eine Veranschaulichung ist in Abbildung 10 gegeben.

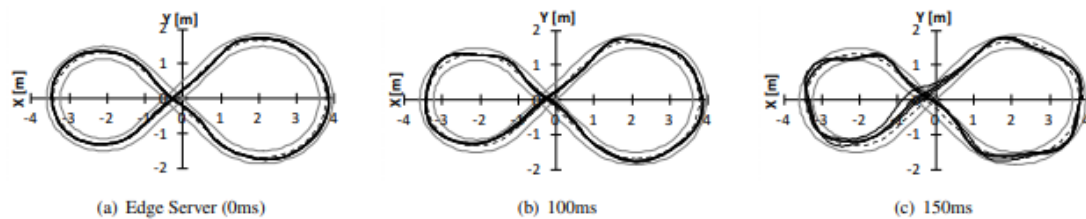


Abbildung 10: Kursabweichung für 0, 100 und 150ms [60]

Dies sind jedoch nicht die einzigen Herausforderungen und Probleme. Was passiert beispielsweise, wenn während einer Fahrt die Verbindung zu den Kartendiensten verloren geht oder man Straßenabschnitte betritt, zu welchen vorher noch keine Daten erhoben und dementsprechend Karten erstellt wurden. Wie würde für solche Fälle der „Plan-B“ aussehen?

Lösungssuche

Zu erkennen ist, dass Herausforderung im Zusammenhang mit dem Umgang der erzeugten Daten und die Beschaffung dieser noch sehr groß sind. Demnach ist zu klären, welche aktuellen Lösungsansätze vorhanden sind und wer die führenden Unternehmen sind. Aktuell gibt es drei große Unternehmen, welche sich mit der Erstellung und Verwaltung von HD-/Navigationkarten beschäftigen und dabei auch unterschiedliche Wege einschlagen. Dazu gehören zum einen TomTom mit dessen „Road-DNA“, HERE und deren „HD Live Map“ und die bekannte Google-Maps Plattform.

TomTom: Der niederländische Hersteller bietet seit 2001 Navigationssysteme für Kraftfahrzeuge an und zählt in Europa seither als Marktführer. Um das autonome Fahren und die genaue Lokalisierung des Fahrzeuges zu ermöglichen, wurde die HD-Karte und TomToms „Road-DNA“ entworfen. Hierdurch werden die Straßen und seine Gegebenheiten aus der Sicht eines Autos erfasst (Abbildung 11). Diese deckt aktuell die Verkehrsstruktur von allen 19 Autobahnen in den Ländern Westeuropas, sowie das gesamte Autobahn- und Schnellstraßennetz auf dem US-amerikanischen Festland ab. Erreicht wurden damit in etwa 360.000 Straßenkilometer, welche dem autonomen Fahren als Grundlage zur Verfügung stehen [61].

Zur Erstellung dieser wird zwischen drei Schichten unterschieden. Der Navigationsschicht zur Berechnung der Fahrtroute, die Lokalisierungsschicht, welche die Position des Fahrzeuges in der Fahrspur bestimmt und der Planungsschicht, welche Daten über Verkehrsschilder, Tempolimits und der Fahrbahngeometrie beinhaltet. Gesammelt und aktualisiert werden die Daten dabei über deren Flottenfahrzeuge.

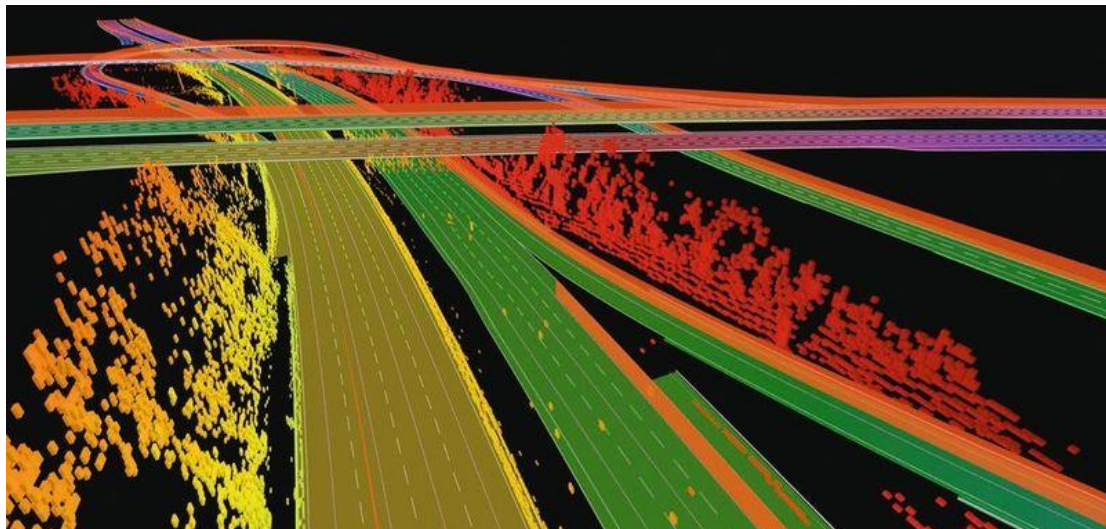


Abbildung 11: HD Map von TomTom [62]

HERE: Der ursprünglich von Nokia für deren Smartphones entwickelte Geodaten und Navigationsdienst HERE verweist ähnliche Erfolge wie dessen Konkurrent TomTom. Dabei besteht jedoch ein großer Unterschied in deren Vorgehensweise, um die Karten zu aktualisieren und zu verwalten. Anders als bei der Konkurrenz wird nicht nur ausschließlich auf bestimmte Flotten, sondern vielmehr auf eine lebende Straße („live road“) gesetzt. Dabei werden über

Schnittstellen sensorbasierte Daten verschiedener Fahrzeuge in die Cloud geladen und verarbeitet. Jedes Fahrzeug, welches mittels seiner Sensoren und Kameras Daten generiert, trägt somit dazu bei, die Karte zu pflegen und eine Echtzeitdarstellung zu ermöglichen. Kurzfristig aufgestellte Baustellen, Stau und vieles mehr kann dadurch in Echtzeit abgebildet werden.

Über die Darstellung des Verkehrs wird es außerdem möglich, virtuelle Fahrten und SIL-Test unter realen Bedingungen durchzuführen. Unterstützt wird dies aktuell von Automarken und Automobilhersteller wie Mercedes, Audi und Hyundai [63].

Google Maps: Der wohl bekannteste Online-Kartendienst wird von dem US-amerikanischen Unternehmen Google LLC mit dessen Google Maps betrieben. Mithilfe von Satellitenaufnahmen und mit Sensoren ausgestattete Fahrzeuge und Menschen werden etwa 98 Prozent der Lebensräume mit Satellitenfotos und etwa 16 Millionen Kilometer Wegstrecke abgebildet [64].

Schon heute bietet er durch die Interaktion und Informationen der Benutzer die Möglichkeit, Gefahrensituationen und Stau zu erkennen oder erhöhtes Verkehrsaufkommen vorherzusagen. Möglich wird dies durch die Nutzung des Navigationsdienstes und das bloße mitführen seines Smartphones.

Trotz dieser gewaltigen Kartierungsmenge eignet sich dieser Dienst nicht zur ausschließlichen Unterstützung von autonomen Fahrsystemen. Der Grund dafür ist die bisherige Nutzung und die Auslegung des Kartendienstes. Erstellt wurden die Karten für die Mensch-Computer Kommunikation. Der Mensch bleibt verantwortlich für das Lenken und Beschleunigen vom Fahrzeug. Deshalb sind die Straßen nicht maßstabsgetreu abgebildet und Verkehrszeichen nicht kartografiert. Dennoch bietet der Dienst vor allem in Hinblick auf seine Reichweite und Aktualisierung großes Potenzial, zur Unterstützung hochgenauer Karten.

Maßnahmen:

Nachdem die Probleme dargestellt und aktuelle Akteure benannt wurden, sollen im Folgenden Maßnahmen vorgestellt werden, wie der Einsatz von Cloud bei der Erstellung, Verwaltung und Nutzung von HD-Karten effizient eingesetzt werden kann. Ziel ist es dabei, eine regenerative Karte wie in Abbildung 12 dargestellt zu schaffen. Die Fahrzeuge sammeln mit deren Sensoren und Kameras Daten während der Fahrt und übertragen diese in die Cloud zur Speicherung.

Map Provider können diese Sensordaten beziehen und mithilfe entsprechender Algorithmen oder Deep-Learning-Verfahren in ihre HD-Karte integrieren, welche anschließend in die Fahrzeuge geladen wird. Dadurch erhalten die Fahrzeuge eine reale Darstellung des Verkehrs, können gefahren und Stausituation umfahren und das Fahrzeug schlussendlich navigieren.

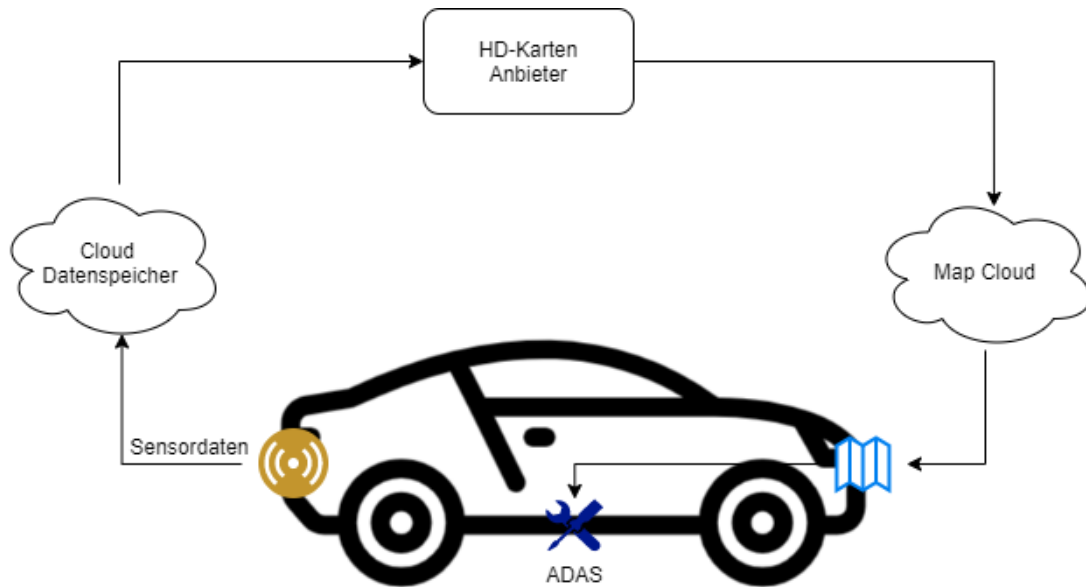


Abbildung 12: Selbstteilende Karte

Der Ansatz hierzu wäre die Nutzung einer hybriden Cloud, wie sie in 2.1.3 bereits beschrieben wurde. Hierdurch wird es möglich, eine Kombination aus public- und private-Cloud zu verwenden. Sensible Daten, welche beispielsweise den Datenschutz betreffen (Kennzeichen, Gesichter) oder unternehmensintern bleiben sollen können somit in der firmeneigenen Cloud betrieben werden, wohingegen Sensorinformationen über Straßen, Verkehr und vieles mehr in einer öffentlichen Cloud HD-Karten Providern zur Verfügung gestellt werden können.

Bei den HD-Karten ist es wichtig zu beachten, dass diese eine Maschine-zu-Maschine Kommunikation ermöglichen sollen, weshalb die Art und Genauigkeit der Darstellung der Daten dementsprechend angepasst werden muss. Demnach ist es nicht mehr wichtig, große und übersichtliche Straßen darzustellen, sondern diese maßstabsgetreu und detailliert zu gestalten. Um dies zu realisieren und die Menge der Daten einordnen und später komprimieren zu können, werden die bezogenen Daten in vier Kategorien unterteilt. Der Navigationsschicht, der Straßengeometrieschicht, der Verkehrsregelschicht und der Echtzeitschicht.

- **Navigationsschicht:** Beinhaltet Informationen über alle Straßen und Abbiegungen. Diese werden als Kanten und Knoten dargestellt, wobei die Straßen als Kanten und Kreuzungen bzw. Schnittpunkte zwischen diesen Straßen als Knoten dargestellt werden. Damit ist eine Routenplanung von A nach B möglich.
- **Straßengeometrieschicht:** Charakterisiert den Verlauf, die Breite und Fahrspuren der Straße, damit das Fahrzeug beispielsweise seine Fahrt und Wendemanöver planen kann. Zudem beinhaltet es Informationen über Brücken und Tunnel.
- **Verkehrsregelschicht:** Impliziert Informationen bezüglich der Möblierung der Straße und seiner Umgebung. Dazu gehören Verkehrszeichen, Ampeln, aber auch signifikante Objekte und Gebäude, welche der Lokalisierung dienen können.
- **Echtzeitschicht:** Während die anderen drei bisherigen Schichten hauptsächlich aus statischen Informationen bestehen, fließen in die Echtzeitschicht hauptsächlich dynamische Informationen ein. Diese können beispielsweise aus der Dichte des Verkehrs, neuen Baustellen, Unfällen oder auch Stauen bestehen.

Diese vier Schichten sollen im Folgenden dazu dienen, die Daten kategorisieren zu können und den Umfang der Daten zu reduzieren. Dabei wird zwischen Verfahren zur Entwicklung und des weiteren Betriebs einer HD-Karte unterschieden.

Entwicklung: Wie zuvor erläutert, fallen bei den Testfahrten zur Sammlung der Straßendaten und Generierung der Maps große Datenmengen an, welche mittels der Sensoren erhoben werden. Dabei geht es in erster Linie um das Sammeln von statischen Daten. Statische Daten sind diese, welche sich in der Regel nur sehr selten ändern. Dazu gehören die Schichten der Navigation, Straßengeometrie und Verkehrsregeln. Diese Daten müssen bei neuen Strecken zunächst über Fahrzeuge gesammelt werden, welche mit entsprechender Sensorik dafür ausgestattet sind. Bei dem Befahren neuer Strecken können dabei Datenmengen von mehreren Gigabytes die Minute und bis zu einem Terabyte pro Stunde anfallen. Die Latenz spielt dabei noch keine große Rolle, da diese Daten zur Erstellung der Karte dienen und daher noch nicht von ADAS oder AD eingesetzt werden.

Die von den Sensoren des Fahrzeuges aufgezeichneten Daten müssen zur weiteren Verarbeitung in den Cloud-Datenspeicher transferiert werden. Dabei besteht die Auswahl zwischen einer typischen Relationalen- oder einer NoSQL-Datenbank. Im Zuge dieser Anwendung wird

jedoch eine NoSQL-Datenbank eingesetzt, da sie im Gegensatz zu einer herkömmlichen relationalen Datenbank eine hohe Skalierbarkeit und Leistungsstärke aufweist. Diese wird nötig sein, um die unterschiedlichen Mengen an Daten speichern zu können und gleichzeitig einen schnellen und zuverlässigen Zugriff zu gestatten.

Die größten Anbieter für NoSQL-Datenbanken in dem Gebiet sind Amazon mit seiner DynamoDB, dicht gefolgt von Microsoft Azure und die Cosmos DB und Googles Cloud Datastore. Alle drei besitzen ähnliche Kompetenzen, was die Bereiche der Zuverlässigkeit, Leistung, Skalierbarkeit und den Preis angeht. Im Cloud-Datenspeicher muss nun eine Kategorisierung der Daten stattfinden. Die gesammelten Daten müssen den vorher bestimmten Kategorien zugeordnet werden. Dazu wird Folgendes getan:

- Verarbeitung der Daten: Fehlerhafte Daten werden aussortiert und die Menge der Daten auf das Nötigste zu reduzieren.
- Gruppieren der Daten: anschließend werden die Daten den zuvor vorgestellten Kategorien zugeordnet.

Die Gruppierung der Daten ist deshalb so wichtig, weil anhand dieser die ADAS und weitere Entwicklungen stattfinden können. Mithilfe der Navigationsdaten können beispielsweise Hersteller wie Google Maps eine verbesserte Navigation der Fahrzeuge von einem Punkt zum anderen entwickeln. Andererseits können Kartendiensthersteller wie TomTom oder Here die Informationen der Straßengeometrieschicht nutzen, um anhand signifikanter Objekte eine Lokalisierung des Fahrzeuges zu ermöglichen.

Nachdem die Daten verarbeitet und analysiert wurden, fließen diese in eine Kartendatenbank ein, von wo aus sie in die Fahrzeuge übertragen und von diesen zur Navigation und vieles mehr genutzt werden kann. Dabei ist eine Standardisierung der Daten nötig. Als Protokoll zur Übertragung eignet sich das Navigation Data Standard. Dies ist ein standardisiertes Format für Kartendatenbanken in Navigationssystemen. Entwickelt wurde es mit dem Ziel, eine Datenbank zu schaffen, von der sich die Akteure der Navigation bedienen können, um die Entwicklung zu beschleunigen und voranzutreiben. Gepflegt wird die Navigationsdatenbank von Automobilherstellern wie BMW und Volvo, von Anbietern für Geodaten und Herstellen von Navigationssystemen wie beispielsweise TomTom [65].

Aktualisierung: Wie zuvor angesprochen gibt es auch die Möglichkeit, erstellte HD-Karten während des Betriebs zu aktualisieren bzw. zu bestätigen. Vor allem dynamische Daten, wie beispielsweise das Verkehrsaufkommen, Stau, Unfälle und Baustellen ändern sich stetig. Demnach müssen auch die HD-Karten änderbar sein. In diesem Fall wird davon ausgegangen, dass für die zu befahrenen Strecken bereits eine HD-Karte vorliegt. Es wird sich nun darauf konzentriert, wie die Strecke aktuell gehalten wird. Das Vorgehen ist dabei ähnlich wie jenes zur Erstellung der Karten mit dem Unterschied, dass die Daten nicht von einem speziellen Fahrzeug gesammelt und ausgewertet werden, sondern von allen Fahrzeugen, die mit Sensoren und Kameras ausgestattet sind. Dies würde in erster Linie die Kosten stark reduzieren und zudem die Karten aktueller halten.

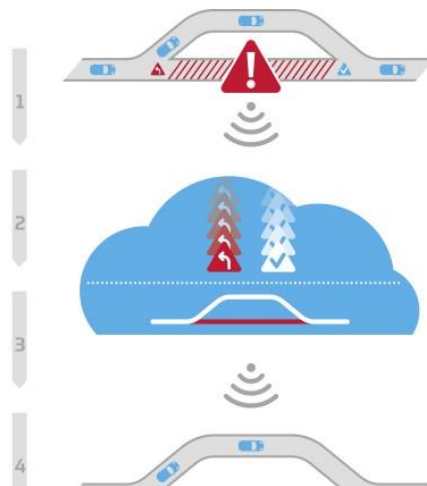


Abbildung 13: Aktualisierung der HD-Karte mithilfe der Cloud [66]

Die von den Fahrzeugen gesammelten Daten müssen zunächst im Fahrzeug komprimiert werden. Dabei werden im Vorhinein unnötige oder fehlerhafte Datensätze aussortiert, um so wenig Daten wie möglich, aber so viele Daten wie nötig zu erhalten. Ein Algorithmus ist nun dafür zuständig, die aufgezeichneten Daten mit denen aus der Cloud bzw. im Speicher des Fahrzeuges zu vergleichen. Wurden Veränderungen an der Straße erkannt, weil beispielsweise eine Baustelle errichtet wurde, dann werden die Daten übertragen. Anschließend finden dieselben Prozesse wie zuvor bei der Erstellung der HD-Karte statt. Die Daten werden in der Cloud verarbeitet, gruppiert und zusätzlich assoziiert.

- Assoziieren der Daten: Hierbei wird überprüft, ob es sich bei den gesammelten Daten beispielsweise um das Schild oder einen bestimmten Straßenabschnitt handelt, welche aktualisiert werden muss.

Nachfolgend wird die HD-Karte aktualisiert und ein entsprechendes Update an die Fahrzeuge versendet. Die Updates werden nicht als ganzes Gebiet oder mit der gesamten Karte, sondern in kleinen Paketen an die Fahrzeuge versendet. Somit wird das Netz zusätzlich geschont und ein zuverlässigerer Transfer, auch an Stoßzeiten, möglich. Durch diese Aktualisierung mittels anderer Fahrzeuge und der Cloud wird es möglich, wie in Abbildung 13 dargestellt, Routen besser zu planen, schneller und umweltfreundlicher an das Ziel zu kommen und Unfälle zu vermeiden.

Übersicht:

Tabelle 2: Übersicht des Anwendungsfalls "Navigation mit Echtzeitkarten"

Anwendungsfall:	Navigation mit Echtzeitkarten
Auslöser:	Das Fahrzeug soll im autonomen Modus mittels der HD-Echtzeitkarte von Punkt A zum Punkt B gesteuert werden.
DB-System:	NoSQL
Cloud-Anbieter:	Amazon: DynamoDB Microsoft: Cosmos DB Google: Cloud Datastore
Echtzeitbearbeitung:	Resultat der zu verarbeitenden Daten muss nicht in Echtzeit zurückgesendet werden. Sofern ein Fahrzeug Veränderungen an der Verkehrsinfrastruktur erkennt, wird die HD-Karte in der Cloud aktualisiert und an Fahrzeuge im selben Gebiet versendet. Hierbei ist eine Verzögerung von einigen Sekunden kein Problem.

Netzwerkausfall:	Umgehen durch Puffern der Daten/Route (ähnlich wie beim Videostreaming). Die Route und dessen Informationen werden im Vorhinein aus der Cloud in den Speicher des Fahrzeuges geladen, um vor Funklöcher gefreit zu sein.
Datenübertragung:	Erstellung neuer Karten: mehrere Gigabyte die Minute Aktualisierung bestehender Karten: max. 15KB/km bei kontinuierlicher Übertragung
Chancen durch die Cloud:	<ol style="list-style-type: none">1. Erstellte HD-Karten können für SIL- und HIL-Tests eingesetzt werden, um Fahrfunktionen, ADAS und weitere Algorithmen kostengünstiger zu testen.2. Effiziente Entwicklung durch „Community information“. Informationen über den Verkehr und seine Infrastruktur werden nicht mehr ausschließlich über spezielle Fahrzeuge gesammelt. Somit kann man schneller und kostengünstiger an Informationen kommen.3. Verwendung verschiedenster Services durch hybrides Cloudsystem und Standardisierung.
Risiken durch die Cloud:	<ol style="list-style-type: none">1. Die Gefahr durch Hackerangriffe steigt. Echtzeitkarten können manipuliert werden, um Fahrzeuge Umwege fahren zu lassen, diese an einem bestimmten Ort abzufangen und zu überfallen oder durch falsche Informationen Unfälle zu verursachen.2. Probleme mit dem Datenschutz. Durch eine ständige Aufzeichnung der Umgebung mittels Sensoren und Kameras können schnell vertrauliche Informationen in die falschen Hände geraten und missbraucht werden.

Bisher wurde ein Konzept beschrieben, durch welches die Navigation mittels HD-Karten und der Cloud ermöglicht und eingesetzt werden kann. Die Informationen hieraus stammen dabei

einzig und allein aus den Sensoren und Kameras der Fahrzeuge. Eine weitere Möglichkeit, um durch den Verkehr zu navigieren, Informationen über den Verkehr zu erhalten und HD-Karten zu verbessern bzw. zu entlasten, kann durch V2X erreicht werden.

3.2.2 V2X am Beispiel Platooning

Bisherige Fahrerassistenzsysteme und KI können hauptsächlich mithilfe der vom Fahrzeug selbst gesammelten Daten und Informationen agieren und Entscheidungen treffen. Dementsprechend kann es vor allem in unübersichtlichen Situationen oder ungünstigen Wetterlagen häufiger zu Fehlentscheidungen kommen [28]. Ein Beispiel hierzu zeigte der Tesla-Unfall, welcher sich am 01. Juni 2020 abspielte. Hierbei liegt ein umgekippter Lastwagen auf der Autobahn und blockierte die Überholspur. Herannahende Fahrzeuge erkannten die Situation und wichen aus. Ein Tesla hingegen, gesteuert durch seinen Autopiloten, erkannte die Situation durch seine Sensoren zu spät und raste in den Lkw [67].

Um solche Unfälle zu vermeiden und die effektive Sichtweite der Fahrzeuge zu erhöhen, konzentrieren sich viele Akteure, darunter Lkw und Automobilhersteller wie Daimler, MAN und Tesla, aber auch Logistikunternehmen wie DHL und DB auf eine weitreichende Kommunikation zwischen Fahrzeugen und ihrer Umwelt. Zusammengefasst wird diese Art der Kommunikation unter V2X (Vehicle-to-everything). Die zwei wichtigsten Techniken sind dabei [68]:

Tabelle 3: Bestandteile des V2X

Vehicle to everything	
V2V: Vehicle to vehicle	Kommunikation zwischen Fahrzeugen, um beispielsweise Gefahrensituationen oder Informationen für den Fahrer oder die ADAS weiterzugeben.
V2I: Vehicle to infrastructure	Kommunikation zwischen Fahrzeugen und Infrastrukturkomponenten (bzw. Road-Side-Units) wie Ampeln, Lampen und Verkehrsschilder.

Die Einsatzgebiete von V2X sind jedoch sehr umfangreich, weshalb sich im Folgenden auf eine spezielle Technik, das Platooning, bezogen wird. Platooning ist ein System für den Straßenverkehr, durch welches es Fahrzeugen möglich wird, in einer Kolonne und sehr dicht hintereinander herzufahren. Der geringe Abstand ermöglicht einen, die Straßenkapazität zu erhöhen, durch einen geringeren Luftwiderstand den Kraftstoffverbrauch zu minimieren und dabei die Verkehrssicherheit nicht zu verringern [69]. Dies geschieht unter den Einsatz von V2V und V2I und ob der Einsatz von Cloud dabei nützlich ist, wird im Folgenden konzipiert.

Problemdiagnose:

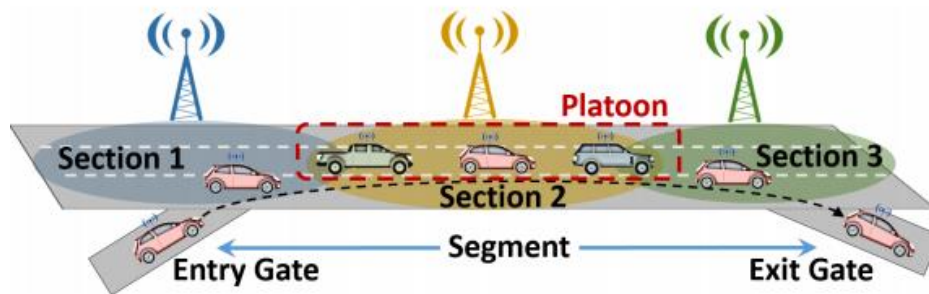


Abbildung 14: Platooning auf der Autobahn [70]

Das Fahren im Platoon (folglich als Zug bezeichnet) ist hauptsächlich für längere Fahrten auf der Autobahn gedacht und soll in erster Linie den Straßengüterverkehr durch Lkw dabei unterstützen, sicherer und effizienter zu fahren. Dabei können bis zu fünf Lkw einen sogenannten Zug bilden, wobei je nach Architektur das erste Fahrzeug als Zugführer die Geschwindigkeit, den Abstand, die Zuggröße und das Ziel festlegt. Mit den darauffolgenden Fahrzeugen findet anschließend ein kontinuierlicher Austausch per V2V-Kommunikation statt, wobei Informationen wie die Geschwindigkeit, das Betätigen der Bremse oder die Fahrtrichtung mitgeteilt werden. Da die Fahrzeuge in einem Zug nur einen sehr geringen Abstand zueinander haben, ist eine hohe Bandbreite, geringe Latenz und hohe Zuverlässigkeit sehr wichtig, um die Vorteile der Technik nutzen zu können, ohne Einbuße in der Sicherheit vorzunehmen [71].

Mit Blick auf die vier ausgearbeiteten Kategorien der Datensammlung aus 3.2.1 wird schnell deutlich, dass beim Platooning, im Gegensatz zu den HD-Karten, ausschließlich mit

dynamischen Informationen gearbeitet wird, welche sich in die Echtzeitschicht einordnen lassen. Dabei gibt es vier große Szenarien, die zu bewältigen sind (dargestellt in Abbildung 14):

Betreten eines Zuges: Bevor ein Fahrzeug sich einem Zug anschließen kann, muss dieser erst mal gefunden werden. Hierbei muss das eigene Fahrzeug feststellen können, wo sich jeweilige Züge befinden, welche Ziele diese ansteuern und ob das eigene Fahrzeug diesen Zug erreichen kann bzw. die notwendige Technik besitzt, um diesen folgen zu können. Ein Problem wird hierbei das Austauschen der Informationen spielen. Wie können die Information über einzelne Züge effizient zwischen den Fahrzeugen auf eine große Distanz übermittelt werden und wie wird trotz dessen der Datenschutz und die Datensicherheit nicht verletzt.

Sofern ein passender Zug gefunden wurde, muss sich das Fahrzeug diesem anschließen können. Um dies zu erreichen, kann das eigene Fahrzeug sich entweder hinten an den Zug ankopeln oder mittels passender Kommunikation zwischen zwei Fahrzeuge einordnen.

Fahren im Zug: Nachdem ein Zug aus Fahrzeugen erfolgreich gebildet wurde, muss dieser in der Lage sein, in Echtzeit untereinander zu kommunizieren. Dazu wird neben einer Kommunikation zwischen den Autos (V2V) auch eine mit der bestehenden Infrastruktur erforderlich sein (V2I). Da die Fahrzeuge sich jedoch auf Autobahnen mit einer Durchschnittsgeschwindigkeit von 80 km/h in ständiger Bewegung befinden, muss unter Umständen mit einer kurzen Verbindungsdauer umgegangen werden können. In der Regel sammelt der Zugführer die Informationen und teilt diese den anderen Fahrzeugen mit. Durch diese Kommunikation wird es anschließend möglich, dass die hinterherfahrenden Fahrzeuge auf Aktionen des Zugführers rechtzeitig reagieren können. Hierbei muss entschieden werden, wie trotz der begrenzten Rechenressourcen der Fahrzeuge die Serviceanwendungen durchgeführt werden können.

Vorbeisicheren an einem Zug: In der Theorie ist die Anzahl der Fahrzeuge, welche sich in einem Zug befinden können, nicht begrenzt. Dennoch führen längere Schlangen von dicht hintereinander folgenden Fahrzeugen zu Problemen, wenn es darum geht, andere Verkehrsteilnehmer an diesem Vorbeifahren zu lassen. Dies kann beispielsweise beim Auf- oder Abfahren der Autobahn passieren. Aus diesem Grund muss die Anzahl der Fahrzeuge in einem Zug begrenzt werden. Das System muss trotz dessen in der Lage sein, anderen Fahrzeugen den Spurwechsel zu ermöglichen, indem eigenständig eine Lücke geschaffen und diese später wieder geschlossen wird. Dieser Vorgang wird neben hochmodernen Sensoren, welche die Fahrzeuge

identifizieren, auch zusätzliche Rechenleistung und Lokalisierung der eigenen Fahrzeuge in Anspruch nehmen.

Verlassen eines Zuges: Für das Verlassen eines Zuges gibt es mehrere Szenarien. Das erste Szenario stellt ein einzelnes Fahrzeug dar, welches den Zug verlassen möchte, weil es beispielsweise seinen Zielort erreicht hat. Handelt es sich dabei nicht um den Zugführer, dann müssen die anderen Fahrzeuge, ähnlich wie beim Vorbeisicheren, genügend Platz für das Fahrzeug schaffen, welches den Zug verlassen möchten. Dazu muss die Umgebung überwacht, andere Fahrzeuge berücksichtigt und gegebenenfalls mit ihnen kommuniziert werden. Anschließend muss die Lücke geschlossen und der freie Platz im Zug für andere Fahrzeuge erkenntlich werden. Sollte der Zugführer jedoch den Zug verlassen, so muss unter den Fahrzeugen ein neuer Zugführer bestimmt werden, welcher sich an die Spitze des Zuges einordnet.

Im anderen Szenario wird der Zug aufgelöst. Dies kann unter Umständen geschehen, wenn der komplette Zug seinen Zielort erreicht hat. Für diesen Fall müsste lediglich ein Abstand der Fahrzeuge im Zug hergestellt werden und anschließend die Kontrolle an den Fahrer übergeben werden.

Die Hauptprobleme des Platooning liegen demnach in der Verarbeitung und Übertragung der Daten. Anders als bei den HD-Karten muss die Verarbeitung der eingehenden Daten direkt und in Echtzeit erfolgen, da andernfalls schwere Unfälle folgen. Interessant ist es deshalb zu sehen, wie bisherige Architekturen aussehen und ob diese Probleme mit der Cloud zu lösen sind.

Lösungssuche:

Die Art der Kommunikation unter den Fahrzeugen und der Infrastruktur und die Ausführung des Platooning wird über verschiedene Wege gehandhabt. Zwei Systeme machen dabei jedoch aktuell von sich die Rede, welche mit LTE und nachfolgend auch im Zusammenhang mit 5G für Konnektivität und Vernetzung sorgen sollen. Gemeint ist hierbei Dedicated Short Range Communication (DSRC) und Cellular Vehicle-to-X (C-V2X).

DSRC: 1999 wurde der DSRC-Technologie, welche im Wesentlichen ein WLAN-System ist, von der US Federal Communications Commission (FCC) das 5,9 GHz-Band zugeteilt, welches

den Dienst über kurze und mittlere Entfernungen effektiv macht und für geringe Latenz und hohe Zuverlässigkeit sorgt. Aufgrund seiner geringen Reichweite treten selbst bei extremen Wetterbedingungen nur geringe Interferenzen auf, womit die Kommunikation zu und von schnell fahrenden Fahrzeugen realisiert werden kann. Diese drahtlose Kommunikationstechnologie umfasst sowohl die V2V als auch die V2I-Kommunikation und wird mittels Transponder umgesetzt, welche mit den jeweiligen On-Board-Units oder Roadside-Units kommunizieren. Dazu sendet jedes Fahrzeug in etwa zehnmal die Sekunde seine wichtigsten Informationen wie Standort, Richtung und Geschwindigkeit an umliegende Fahrzeuge, welche anhand dieser Informationen aufkommende Risiken einschätzen können [72] [73].

C-V2X: Als Gegenspieler der DSRC setzt sich die deutlich jüngere C-V2X-Technologie immer mehr durch. Sie verfolgt denselben Zweck der direkten Kommunikationsverbindung zwischen Fahrzeugen untereinander und mit bestehender Infrastruktur mit dem Unterschied, dass diese sowohl mit als auch ohne zugrundeliegenden Mobilfunknetz funktioniert. Dabei wird zwischen zwei Übertragungsmodi unterschieden. Der Erste stellt eine V2V, V2I und eine Kommunikation zwischen Fahrzeug und Fußgänger/Radfahrer her, welche unabhängig vom Mobilfunknetz funktioniert. Der zweite Modus ermöglicht den Fahrzeugen über das herkömmliche Mobilfunknetz Informationen über Straßenverhältnisse und den Verkehr in der Umgebung zu empfangen [74]. Der gesamte Ansatz ist somit komplexer und technologisch anspruchsvoller. Zudem wird es voraussichtlich unumgänglich sein, dass Fahrzeuge umgebaut und beispielsweise um ein zweites LTE-Modem erweitert werden müssen [75].

Um anschließend die Vision vom gewünschten Platooning umzusetzen, wird häufig auf funktionale Architekturen zurückgegriffen. Diese bestehen in der Regel aus unterschiedlichen Schichten, welche sowohl auf bordeigene Fahrzeugsysteme als auch auf straßenseitige Infrastrukturen wie Server und die Cloud verteilt sind. Das früheste Beispiel einer Platoon-Architektur wurde mit dem PATH-Programm (partners for advanced transit and highways) schon in den 80ern vorgeschlagen. Das Ziel ist es, die Autobahnkapazität mit so wenig Veränderung an der Infrastruktur wie nur möglich zu erhöhen. Dazu gibt es folgende Schichten [70]:

- **Physikalische:** Regelt die Steuerungsmodulare für das Fahrzeug. Betroffen sind Elemente wie die Lenkung, Bremsen, Getriebesteuerung und lokale Sensoren.

- **Regulatorische:** Plant und führt lokale Manöver aus, welche von der oberen Schicht benötigt werden.
- **Koordinierende:** Plant mithilfe der Informationen und Unterstützung anderer Fahrzeuge und der Infrastruktur auszuführender Manöver.
- **Verbindungs- und Netzwerkschicht:** Steuerung vom Verkehrsfluss. Dabei dient die Verbindungsschicht zur Steuerung von Autobahnabschnitten, wohingegen die Netzwerkschicht den Verkehrsfluss über das gesamte Autobahnnetz plant, um die Kapazität zu maximieren und die Fahrzeit zu minimieren.

Mit der Zeit haben sich auch andere Architekturen durchgesetzt. Die größten Unterschiede liegen dabei in der Art zur Steuerung des Zuges und der Berechnung intensiver Aufgaben für das System. Wo über das PATH-Programm der gesamte Zug durch das erste Fahrzeug koordiniert wird, gibt es auch Ansätze, welche die Zugentscheidungen unter den Zugmitgliedern aufteilen und somit für eine Verteilung der Rechenaufgaben sorgen. Auch Lösungen mit Cloud Computing Frameworks setzen sich immer mehr durch, darunter auch beispielsweise CARMA (Cloud-Assisted Real-time Methods for Autonomy), zur Unterstützung bestehender Architekturen und Erweiterung der Berechnungsfähigkeit [68] [70].

Maßnahmen:

Nachdem die Probleme bezüglich der Latenz und Rechenintensität dargestellt und aktuelle Akteure sowie Techniken benannt wurden, sollen im Folgenden Maßnahmen vorgestellt werden, welche die Entwicklung vorantreiben sollen. Dabei wird vor allem ein Fokus auf die Cloud gelegt und inwiefern sich diese in den Prozess integrieren lässt. Das Ziel ist dabei die Optimierung der Platooning-Technologie, welche unter anderem durch V2V-, V2I-Kommunikation und einer zuverlässigen Echtzeitverarbeitung die Autobahnen entlasten und den Verkehr dort sicherer gestalten soll. Wichtig ist jedoch die Annahme, dass man sich auf einer gemischten Autobahn befindet. Dies bedeutet, dass sich neben den Fahrzeugen, welche mit entsprechender Technik für die V2X-Kommunikation wie Antennen, WLAN-Module und Anzeigen für den Fahrer ausgestattet sind, auch ältere Fahrzeuge bewegen, welche über eben diese Technik nicht verfügen. Somit wird es nicht möglich sein, mit jedem Fahrzeug direkt zu kommunizieren, um beispielsweise Manöver durchzuführen oder Fahrzeuge passieren zu lassen.

Mit dieser Erkenntnis kann eine Optimierung für die zuvor erhobenen Probleme erarbeitet werden. Dabei ist der Einsatz der Cloud jedoch nur bedingt sinnvoll. Der Grund hierfür liegt in der Latenz, welche für das Platooning immens wichtig ist. Sie bestimmt, wie schnell die Daten von der Quelle zum Ziel gelangen, wodurch schnelle Manöver berechnet oder Informationen von Infrastrukturen erhalten werden können. Sie lässt sich durch mehrere Faktoren beeinflussen. Darunter die physische Entfernung, welche die Datenpakete zwischen den jeweiligen Servern und dem Endgerät zurücklegen müssen und die Menge des Datenverkehrs im Netz. Mit zunehmender Autonomie und Digitalisierung nimmt der Datenverkehr stetig zu und auch die Entfernungen zwischen Cloud-Servern und Fahrzeugen und deren Geschwindigkeit auf Autobahnen sind ein ausschlaggebender Grund dafür, dass bei einem 4G-Netz die durchschnittliche Latenz zwischen Fahrzeug und Cloud bei $>80\text{ms}$ liegt [76].

Eine Lösung zur Überwindung der Latenz könnte eine verwandte Technologie des Cloud Computing sein, welche das Rechenzentrum näher an die Geräte bringt. Die Rede ist von Edge-Clouds oder auch dem Edge-Computing. Hierbei findet die Verarbeitung, Speicherung oder Analyse der Daten nicht wie bei einer Cloud zentral, sondern dezentral an jeweils eigenen Rechenzentren statt [77]. Diese befinden sich optimalerweise verteilt an Verkehrsinfrastrukturen und ermöglichen somit eine schnellere und zuverlässige Übertragung der Daten.

In der nachfolgenden Architektur wird eine Lösung aufgezeigt, mit welcher sich sowohl der Vorteil der Edge-Server als auch die Leistung der Cloud-Server integrieren lassen. Dabei wird zwischen drei Ebenen unterschieden, wobei die letzte Ebene bezüglich der Cloud und der daraus ergebenden Chancen genauer betrachtet wird.

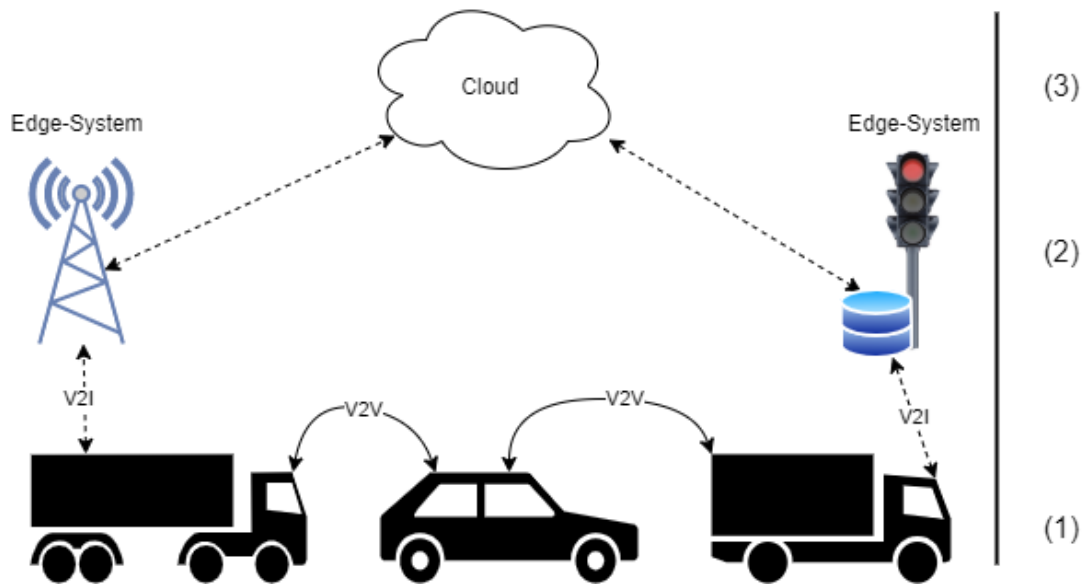


Abbildung 15: Platooning-Architektur (Vehicle-to-Edge-to-Cloud)

- (1) In der ersten Ebene findet die zuvor beschriebene V2-Kommunikation statt. Hierbei tauscht sich der Zug untereinander aus und kommuniziert zusätzlich mit Infrastruktur in der Nähe.
- (2) Am Rand des Netzes befinden sich die Edge-Systeme. Diese können aus Mobilmasten oder herkömmlichen Infrastrukturen bestehen, welche speziell mit entsprechender Technik ausgerüstet wurden. Ihre Aufgabe ist es, relevante Anfragen der Fahrzeuge direkt zu bearbeiten und an umliegende Fahrzeuge weiterzugeben.
- (3) Größere Aufgaben oder jene, welche keine Echtzeitbearbeitung benötigen können an die Cloud weitergeleitet werden. Dadurch werden Edge-Server entlastet und der Vorteil der Cloud ausgenutzt.

Zuvor wurden in der Problemdiagnose vier Kernaufgaben für das Platooning vorgestellt. Diese setzen sich zusammen aus dem Betreten bzw. auch das Finden eines Zuges, das Fahren im Zug, das Vorbeisicheren an einem Zug und das schlussendliche Verlassen eines Zuges. Viele dieser Probleme werden in den ersten beiden Ebenen aus Abbildung 15 angegangen. Um einen Zug bilden zu können oder einem beizutreten, müssen die Fahrzeuge über entsprechende Technik verfügen. Demnach kann das Betreten über die erste Ebene stattfinden. Mittels einer V2V Kommunikation teilt das Fahrzeug dem Zug mit, dass er sich diesem anschließen möchte und

die nötigen Vorkehrungen werden diesbezüglich getroffen. Das anschließende Fahren im Zug und das Vorbeisicheren anderer Fahrzeuge wird über die ersten beiden Ebenen koordiniert. Der Zug oder der Zugführer erhalten durch die Infrastruktur Informationen über die Verkehrslage und können sich dementsprechend auf diese vorbereiten.

Ein Beispiel dazu wäre es auf eine Baustelle zuzufahren. Einige Kilometer vor der Baustelle können Warnschilder mit dementsprechenden Edge-Systemen ausgestattet sein. Diese beinhalten die Informationen über die Baustelle, mögliche Stausituationen oder auch Straßengegebenheiten, welche sie unter anderem auch durch die vorausfahrenden Fahrzeuge erhalten. Ein näherer Zug würde allein durch seine Sensoren die drohende Gefahr erst zu spät erkennen können, deshalb wird dieser mittels V2I über die Gegebenheiten informiert. Mithilfe von V2V kann sich der Zug anschließend koordinieren, sein Tempo verlangsamen und den Abstand ggf. wiederherstellen. In diesem Szenario wäre es für die Warnschilder, aber auch die Fahrzeuge unpraktisch, Daten über ein Fernnetz zurück an einen zentralen Cloud-Kern zu senden.

Auch das Verlassen und Auflösen eines Zuges kann größtenteils ohne die Cloud realisiert werden. Stellt sich jetzt also die Frage, bei welchen Prozessen die Cloud von Nutzen wäre und den Prozess der Entwicklung vorantreiben würde. In erster Linie liegt die Aufgabe der Cloud in diesem System in der Vor- und Nachbearbeitung.

Vorbearbeitung: Ein wichtiger Prozess, bei dem die Cloud einen nützlichen Beitrag leisten würde, wäre zum Finden bestehender und Bilden neuer Züge. Eine dynamische Zuordnung von Fahrzeugen und bestehenden Zügen in der Nähe kann dabei über die Cloud stattfinden. Hierzu können Züge und einzelne Fahrzeuge, welche einem Zug beitreten oder selbst bilden wollen, als ID mit deren Geschwindigkeit, Position und Ziel in einer IoT-Cloud hinterlegt und dargestellt werden. Die Position kann wie in 3.2.1 beschrieben ermittelt werden und den Fahrzeugen dabei helfen, Züge auch über größere Entfernungen zu finden und diesen beizutreten.

Dank der Kommunikation zur Cloud kann somit zusätzlich für mehr Sicherheit in Bezug auf den Datenschutz und zur Vermeidung von Manipulation gesorgt werden. Anfragen in Bezug auf Platooning können vorher überprüft, anonymisiert und böswillige Systeme blockiert werden.

Nachbearbeitung: Nachdem Züge gebildet wurden, müssen diese weiterhin mit Informationen über die Straßen und Verkehrslage versorgt werden. Ein großer Teil davon findet, wie oben beschrieben, durch die Infrastruktur, Edge-Systeme und einzelne Fahrzeuge statt. Jedoch kann dies auch über eine Zentrale Cloud laufen. Diese kann beispielsweise von der Polizei, Rettungskräften oder Behörden verwendet werden, um Gefahrenmeldungen über bestimmte Bereiche zu versenden oder Empfehlungen auszusprechen. Ein weiteres Einsatzgebiet der Cloud für bestehende Züge wäre die nachträgliche Bearbeitung von Informationen. Nachdem durch die fahrzeugeigenen Rechensysteme oder Edge-Systeme Manöver oder der optimale Mindestabstand der Fahrzeuge berechnet wurde, können diese Informationen zusätzlich von der Cloud berechnet werden. Somit erhalten die Fahrzeuge erste zuverlässige Ergebnisse in Echtzeit durch Edge-Systeme, welche nochmals nachträglich mit einer Verzögerung von der Cloud bestätigt oder korrigiert werden.

Schlussfolgernd ist zu erkennen, dass der ausschließliche Einsatz der Cloud für ein System wie das Platooning aufgrund seiner Entfernung und der daraus folgenden Verzögerung ungeeignet ist, jedoch für Aufgaben im Hintergrund und als Ergänzung anderer Systeme durchaus sinnvoll eingesetzt werden kann.

Übersicht:

Tabelle 4: Übersicht des Anwendungsfalls "V2X am Beispiel Platooning"

Anwendungsfall:	V2X für Platooning
Auslöser:	Mehrere Fahrzeuge möchten sich zusammenschließen, um gemeinsam als Zug über die Autobahn zu fahren und somit die Sicherheit zu erhöhen und den Kraftstoffverbrauch zu senken.
Cloud-Systeme:	Berechnung für Manöver und den Zug: Über Cloud Compute Funktionen. Darstellung der Fahrzeuge und Züge: IoT – Cloud zur Vernetzung der Fahrzeuge und Infrastrukturen

<p>Echtzeitbearbeitung:</p>	<p>Eine Echtzeitverarbeitung der Daten ist für dieses System nötig, um Manöver sicher durchzuführen, über die Verkehrslage informiert zu sein und ggf. auf andere Verkehrsteilnehmer reagieren zu können. Eine Verzögerung kann für ein zu spätes Reagieren oder Einlenken der Systeme und somit zu schweren Unfällen führen.</p>
<p>Netzwerkausfall:</p>	<p>Eine Verbindung mittels V2V, V2I (zu den Edge-Servern) oder zur Cloud muss gewährleistet sein, damit das System ausgeführt werden kann. Eine kurze Unterbrechung zu einem dieser drei Kommunikationsarten kann dabei jedoch durch die anderen beiden ausgeglichen werden. Sollte beispielsweise aufgrund der Geschwindigkeit und Position des Fahrzeuges kurzzeitig die Verbindung zu Edge-Servern in der Nähe unterbrochen werden, kann die Geschwindigkeit verringert und die fehlenden Informationen kurzfristig über V2V oder der Cloud nachgeholt werden. Bei einem längeren Ausfall muss der Zug jedoch aufgelöst werden.</p>
<p>Chancen durch die Cloud:</p>	<ol style="list-style-type: none"> 1. In Zusammenarbeit mit HD-Karten (3.2.1) können Züge schneller und effizienter gefunden und gebildet werden. 2. Einsatz zur Ergänzung von Navigationsdiensten. 3. Nachträgliche Bearbeitung des empfohlenen Mindestabstandes der Züge. 4. Behörden und staatliche Organisationen können schneller Warnungen an die Verkehrsteilnehmer senden, als es aktuell durch das Radio der Fall ist.
<p>Risiken durch die Cloud:</p>	<ol style="list-style-type: none"> 1. Zu große Latenzen und Übertragungszeiten können zu Sicherheitsmängeln führen. 2. Nachträgliche Bearbeitung der Informationen kann zu einer Fehleinschätzung des Systems führen, da der Zug sich in der Bearbeitungszeit weiterhin fortbewegt und ggf. eine andere Ausgangssituation hat.

3.2.3 Smart Parking

Während die Städte, deren Einwohner und damit auch die Anzahl der Autobesitzer immer weiter steigen, sinken die Zahlen der freien Parkplätze in den Großstädten stetig [78]. Die Folge daraus ist eine lange Parkplatzzsuche, welche in der Regel 7-8 Minuten in Anspruch nimmt. Dies beeinflusst nicht nur indirekt die Lebensqualität, sondern sorgt zudem noch für Stau in den Städten, erhöhte Luftschadstoff- und Lärmemissionen und ein erhöhtes Unfallrisiko [79]. Bemühungen, mehr Parkraum zu schaffen und die Städte dementsprechend anzupassen, sind meistens mit großen Kosten verbunden.

Um ein schnelleres Finden freier Parkplätze zu ermöglichen und dadurch weniger Stress, mehr Zeit und größere Kraftstoffersparungen zu erreichen, setzen viele Städte wie beispielsweise auch Hamburg auf das „Smart Parking“ [80] [81]. Hierbei gibt es zwei Ansätze, wie freie und belegte Parkplätze identifiziert werden:

1. **Dynamisch:** Sensoren von modernen Fahrzeugen und fest installierte Kameras werden verwendet, um mittels Deep-Learning Verfahren freie und belegte Parkplätze zu erkennen und diese Information zur Verfügung zu stellen.
2. **Statisch:** Eingebaute Sensoren in den Böden der Parkplätze überprüfen den Status des jeweiligen Parkplatzes. Diese befinden sich meistens in Parkhäusern oder auf stark befahrenen Straßen und sind in der Regel kostspieliger.

Aus den bisherigen Ergebnissen der Arbeit in 3.2.1 und 3.2.3 können wir schlussfolgern, dass die Cloud im Kontext der automobilen Entwicklung weniger für Echtzeitverarbeitungen und -dienste eingesetzt werden kann und dafür einen großen Nutzen in der Haltung, der Aufbereitung und der Verfügbarkeit und Bereitstellung der Daten erbringt. Im Folgenden wird sich auf den dynamischen Ansatz bezogen, welcher wie in Abbildung 16 dargestellt die Parkraumtopologie mittels Fahrzeuge und Kameras erfasst und zur Verfügung stellt. Der Nutzen der Cloud soll diesbezüglich, im Hinblick des Smart Parking, konzipiert werden.

Problemdiagnose:

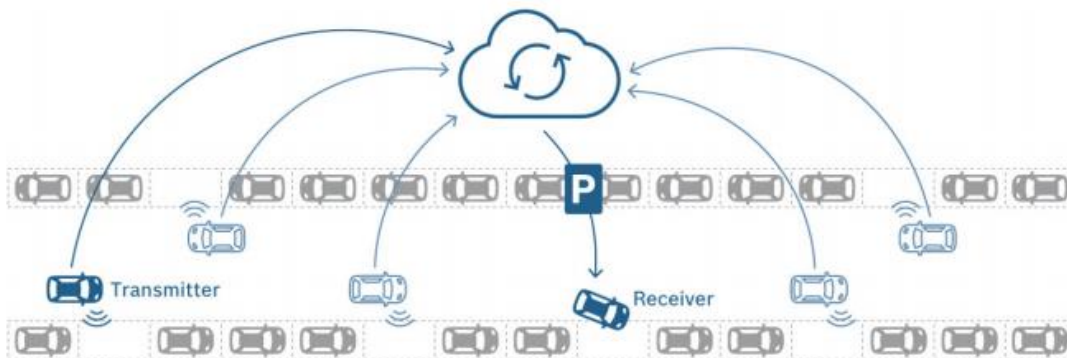


Abbildung 16: Smart Parking Prinzip [55]

Einer für alle und alle für einen ist das Motto vom Smart Parking. Dabei soll es Autofahrern möglich werden, in nahezu Echtzeit freie Parkplätze in der Umgebung angezeigt zu bekommen, damit die lange Suche nach Parkplätzen und die damit verbundenen Probleme reduziert werden. Abbildung 16 zeigt das Prinzip, welches dahintersteckt. Die Parkplätze, beispielsweise am Straßenrand, werden während einer normalen Fahrt von den Sensoren moderner Fahrzeuge auf freie Parkplätze analysiert. Dazu können die Daten mit bestehenden Daten verglichen oder ggf. neue Parkraumkarten erstellt werden. Wurde ein gültiger und freier Parkplatz gefunden, muss dieser anschließend einer Datenbank hinzugefügt und den teilnehmenden Autofahrern zur Verfügung gestellt werden. Somit kann eine dynamische Belegungskarte der Umgebung erstellt und Autofahrern den Weg zu freien Parkplätzen gezeigt werden [82].

Die Hauptherausforderungen, welche sich dabei ergeben, lassen sich auf drei Grundfunktionen ableiten. Das Erkennen freier und gültiger Parkplätze, die Bereitstellung freier und gültiger Parkplätze und das Lenken bzw. Navigieren zu diesen [83].

Erkennen freier und gültiger Parkplätze: Neben der großen Herausforderung zur Unterscheidung zwischen freien und belegten Parkplätzen wird das Erkennen realer und gültiger Parkplätze eine große Hürde sein. Gültige Parkplätze müssen klar von Parkverbotszonen, Bushaltestellen oder Grundstücksausfahrten getrennt werden, damit die Fahrer nicht zu falschen Orten navigiert werden, wodurch das System seine vorgestellten Vorteile verlieren würde. Dazu wird es nötig sein, riesige Mengen an Daten zu Analysieren und beispielsweise mithilfe

von Lernalgorithmen eine Aussage über die Gültigkeit der jeweiligen Parkplätze fällen zu können [82] [84]. Für die Cloud bedeutet dies, dass diese in der Lage sein muss, riesige Datenmengen kostengünstig und schnell zu verarbeiten, damit die Daten über freie Parkplätze dem System rechtzeitig hinzugefügt werden können, bevor diese wieder belegt sind.

Bereitstellen freier und gültiger Parkplätze: Sobald freie und gültige Parkplätze erfasst wurden, müssen diese anschließend verwahrt und anderen Autofahrern zur Verfügung gestellt werden können. Dazu wird es nötig sein, neben der exakten Lokalisierung des Parkplatzes auch die Parkraumtopologie der Umgebung bereitzustellen. Zudem wird auch die Aktualität der Daten eine wichtige Rolle spielen. Da man nicht immer davon ausgehen kann, dass alle Fahrzeuge mit dementsprechenden Sensoren und Techniken ausgestattet sind, um aktuelle Belegungen der Parkplätze anzeigen zu können, könnten durchaus besetzte Parkplätze weiterhin als frei angesehen werden [82]. Schätzungsweise müssen 1,5 % der Fahrzeuge im Verkehr mit dementsprechender Technik ausgestattet sein, damit das System effizient funktioniert und selbst dabei wird es beispielsweise auf wenig befahrenen Straßen zu Fehlinformationen kommen. Deshalb wird neben der Bereitstellung der erfassten Parkplätze auch eine Prognose zur Belegung dieser nach einer bestimmten Zeit notwendig sein. Um dies zu erreichen, müssen im Vorfeld Vorhersagen auf Basis bisheriger Informationen getroffen werden können, welche zusätzliche Rechenleistung beanspruchen werden [80].

Navigieren zu Parkplätzen: Nachdem Autofahrer einen freien Parkplatz angefordert haben und ein passender in der Umgebung gefunden wurde, muss dieser abschließend zu dieser Parklücke navigiert werden. Die Herausforderung wird dabei sein, das System mit bestehenden Navigationssystemen zu koppeln oder ggf. über Schnittstellen den Zugriff auf die Position gewünschter Parklücken zu ermöglichen. Dabei sollte möglichst wenig Interaktion manuell durch den Fahrer selbst stattfinden, damit durch Ablenkung keine gefährlichen Situationen entstehen.

Die Hauptherausforderungen beim Smart Parking liegen demnach in der Verarbeitung der Sensordaten zur Identifizierung der Parkplätze und in der Haltung und Bereitstellung dieser Daten für die jeweiligen Fahrer. Im Gegensatz zu den HD-Karten und dem Platooning können ältere Daten ausgewertet werden und zur Ergänzung des Systems bei Ausfall oder Ähnlichem beitragen.

Lösungssuche:

Wie zuvor bereits beschrieben, besteht das Problem mit den fehlenden Parkplätzen vor allem in den Großstädten nicht erst seit gestern. Dementsprechend tummeln sich im Bereich der Parkplatzdienste viele verschiedene Akteure und Stakeholder, was den Markt stark fragmentiert. Die wohl vielversprechendsten Lösungen werden durch INRIX und Bosch vorgeschlagen.

Bosch: Mit dem Serienstart des Community-based Parking begann das deutsche Unternehmen 2018 der Suche nach freien Parkplätzen in Großstädten den Kampf anzusagen. Dabei werden die in den meisten modernen Fahrzeugen bereits seit Werk integrierten Parksensoren und Vernetzungshardware verwendet, um Parklücken zu erfassen und diese Daten verschlüsselt und anonymisiert an deren Bosch IoT Cloud zu senden. In der Cloud werden die Daten gesammelt, sortiert und anhand von Algorithmen interpretiert. Mithilfe bordeigener Navigationssysteme oder Smartphones können Autofahrer anschließend freie Parkplätze anfragen und zu diesen geleitet werden. Der Service wird jedoch nicht nur als Echtzeitkarte für Parkplätze verwendet, sondern bietet darüber hinaus auch die Chance, Prognosen und Informationen zu Parkplätzen zu erhalten.

Zukünftig plant das Unternehmen anhand dieser Informationen autonome Fahrzeuge eigenständig Parken zu lassen. Somit kann der Fahrer bei einem Restaurant besuch beispielsweise vor dem Geschäft halten, aussteigen und das Fahrzeug fährt allein zu einer geeigneten Lücke und auf Wunsch wieder zurück zum Lokal [85].

INRIX: Dem Community-based Parking steht der On-Street Parking Service von Inrix entgegen, welcher in 16 deutschen und nordamerikanischen Städten seit 2017 zur Verfügung steht. Dieser bietet eine Datenbank mit über 35 Millionen Parkplätzen an und verfolgt dabei ähnliche Ansätze wie sein Konkurrent. Die Sensoren der Fahrzeuge werden dazu verwendet, freie Parklücken zu erfassen und mithilfe von Algorithmen und IoT den Autofahrern zur Verfügung zu stellen. Dazu wird außerdem mit Navigationsdiensten wie denen von HERE zusammengearbeitet, wodurch die Parkraumtopologie schneller erfasst und somit zum Erfolg führen kann [86] [87].

Die Lösungen sind in vielen Hinsichten identisch und auch die Cloud wird als Lösungsansatz bereits in Erwägung gezogen. Interessant ist es deshalb herauszufinden, welches Cloud-System für dieses Projekt geeignet ist.

Maßnahmen:

Nachdem die Probleme bezüglich der Verarbeitung und Bereitstellung der Sensordaten vorgestellt und aktuelle Akteure, welche sich an der Entwicklung von Smart Parking beteiligen, benannt wurden, sollen resultierend darauf zwei Cloud Anbieter bezüglich ihrer Kosten und Leistungsfähigkeit miteinander verglichen werden, um den geeigneteren der beiden Anbieter identifizieren zu können. Dazu werden die IoT-Cloud-Anbieter Amazon mit seinem AWS IoT Core und Googles Cloud IoT Core im Verhältnis zueinander gestellt. Beide zählen im Bereich Cloud und IoT als größte Anbieter und sind leistungsfähig genug, um Millionen von Geräten zu verwalten, was im Zuge des Smart Parking nötig sein wird [88].

IoT (Internet of Things) ist ein Netzwerk von verschiedenen Geräten, die mit dem Internet verbunden sind und untereinander interagieren und kommunizieren können. Dadurch ist es möglich, Beteiligungen von Menschen an regulären Vorgängen auszuschließen. Die von den Geräten erzeugten Daten müssen anschließend verarbeitet, interpretiert und gespeichert werden können [89] [90]. In unserem Fall des Smart Parking bedeutet dies, dass Fahrzeuge und deren Sensoren Daten über deren Umgebung erfassen, diese verarbeitet und anschließend in Form einer Parkkarte bereitgestellt und gelagert werden müssen.

Leistungsfähigkeit: Im Allgemeinen ist im Vorhinein zu erwähnen, dass Leistungsangebote und Services zwischen den großen Cloud-Anbietern, zu welchen AWS, Microsoft Azure und Google gehören, recht identisch sind. Jedoch gibt es kleine Unterschiede zwischen diesen Anbietern, welche die Wahl zu einem der beiden ggf. beeinflussen würde.

Beide Anbieter unterstützen als Kommunikationsprotokoll HTTP und MQTT. Konzentriert man sich auf das MQTT Protokoll, wie es in der Studie [89] getan wurde, welche die Servicezeit der beiden Anbieter unter verschiedenen Bedingungen testet, dann wird deutlich, dass sich beide Anbieter bei geringer Datenübertragung in Bezug auf die Latenz sehr ähnlich verhalten. Betrachtet man jedoch den Dateneingang und -ausgang bei steigender Datenübertragung und

verbundenen Geräten, dann ist zu erkennen, dass sich AWS deutlich stabiler im Fan-in Szenario verhält, wohingegen Google eine bessere Performance im Fan-out Szenario abliefert.

Für das Smart Parking wird es voraussichtlicher wichtiger sein, Sensordaten schnell und zuverlässig von den jeweiligen Fahrzeugen und Kameras zu übertragen, damit diese schnell verarbeitet und der Parkdatenbank hinzugefügt werden können. Kurze Verarbeitungs- und Übertragungszeiten bezüglich des Findens und Bereitstellen eines freien Parkplatzes für Autofahrer stellt dabei kein großes Problem da, zumal Anfragen nach freien Parkplätzen seltener geschehen, als dass neue und belegte Parkplätze von Autofahrern identifiziert werden. Aufgrund dessen eignet sich der AWS IoT Core für diese Leistung mehr.

Kosten: Beide Anbieter rechnen nach dem Pay-as-you-go-Prinzip. Googles IoT Core berechnet dabei nach dem monatlich genutzten Datenvolumen ab, wobei für das Anlegen, Auslesen, Aktualisieren oder auch Löschen von Geräten keine extra Gebühren verlangt werden. Amazon hingegen berechnet seine Kosten je nach Region, Gesamtzeit der Verbindungsminuten und der übertragenen Daten. Bei beiden sinken die jeweiligen Kosten, zum Beispiel für die Datenübertragung, mit Anstieg der ausgetauschten Daten [89]. Um die Kosten für die Datenübertragung im Folgendem grob einschätzen zu können, wird davon ausgegangen, dass jede Minute 1 kB an Daten jeweils versendet werden. Mit Blick auf Abbildung 17 wird dabei deutlich, dass die Kosten für wenige verbundene Geräte bei Google deutlich höher sind als bei AWS. Mit Zunahme der verbundenen Geräte nimmt demnach auch der Datenaustausch und infolgedessen die Kosten zu. Bei Google fallen diese jedoch aufgrund des Preismodells zum Ende hin geringer aus als bei AWS.

Devices	AWS	GCP
1-6	<15	Free
7-70	<3	<10
70-250	3-15	10-45
250-1000	15-56	45-185
1000-4100	56-230	185-810
4100-10000	230-560	810-1440
10000-50000	560-2500	1440-4640
50000-100000	2500-4800	4640-8640
100000-420000	4800-17700	8640-16300
420000-500000	17700-21058	16300-17815

Abbildung 17: Preisveränderung in \$ für AWS und Google pro Monat [89]

Der Grund dafür ist, dass ab einer monatlichen Datenübertragung von > 5 TB der einzelne Preis bei 0,00045 \$ pro MB liegt, wohingegen der Preis bei AWS in der Region Frankfurt ab über fünf Milliarden versendeter Nachrichten bei 0,84 \$ pro Millionen Nachrichten liegt. Hinzu kommt noch, dass bei AWS die Nutzung von jedem weiteren Service separat beansprucht werden muss. Google hingegen bietet dahingegen eine Out-of-the-box-Funktion, welche Datenanalyse oder Machine-Learning-Dienste einschließt. Damit das Smart Parking funktioniert, müssen viele Geräte, darunter Fahrzeuge, Sensoren oder auch Kameras, miteinander verbunden sein. Deshalb würde sich dahingehend Googles IoT Core als kostengünstigere Lösung eignen [91] [92] [93].

Am Ende ist es jedoch schwierig zu sagen, welcher Anbieter sich als geeigneter herausstellt, weil beide ihre Vor- und Nachteile besitzen. Im Bezug auf die bisherigen Erkenntnisse und mit Blick auf die jeweiligen Partner, welche in [94] und [92] genannt werden, würde AWS gegenüber Google Cloud dominieren.

Überblick:

Tabelle 5: Übersicht des Anwendungsfalls "Smart Parking"

Anwendungsfall:	Smart Parking
Auslöser:	Mit Sensoren ausgestattete und dem Internet verbundene Fahrzeuge identifizieren durch das Vorbeifahren freie Parkplätze und stellen diese anderen zur effizienteren Parkplatzsuche zur Verfügung.
Cloud-Anbieter:	IoT Cloud-Anbieter wie AWS IoT Core oder Googles IoT Cloud
Echtzeitverarbeitung:	Parkplätze sollten in nahezu Echtzeit zuverlässig identifiziert und bereitgestellt werden, damit Fahrzeuge nicht zu veralteten und bereits besetzten Parkplätzen geführt werden.
Netzwerkausfall:	Ein Ausfall des Systems oder Netzwerkes würde eine Aktualisierung der Parkplätze behindern, das System jedoch nicht unbrauchbar machen, so wie es bei Navigation mit Echtzeitkarten

	und V2X am Beispiel Platooning der Fall wäre. Prognosen bezüglich des Zustandes können weiterhin gegeben werden.
Chancen durch die Cloud:	<ol style="list-style-type: none">1. Eine Datenbank in der Cloud ermöglicht es zusammen als Community Parkplätze schneller zu finden und zu identifizieren.2. Daten können über die Cloud oder als hybride Cloud mit HD-Karten geteilt werden, um beide Systeme effizienter zu machen.3. Sammeln von Parkplatzinformationen, um langfristig für seltener befahrene Straßenabschnitte zuverlässige Prognosen aufstellen zu können
Risiken durch die Cloud:	<ol style="list-style-type: none">1. Veraltete Parkplatzinformationen können Fahrer fehlleiten und somit Stau und Kraftstoffverbrauch erhöhen.

4 Prototypische Umsetzung

Nachdem im letzten Kapitel spezifische Anwendungsfälle für ADAS/AD beschrieben und die jeweiligen Möglichkeiten zur Verbesserung mithilfe der Cloud konzipiert wurden, wird im Folgenden anhand eines dieser Szenarien der Einsatz einer Cloud prototypisch umgesetzt. Dazu wird der Anwendungsfall V2X am Beispiel Platooning durch die Google Cloud SQL technisch untersucht und angewendet. Wie in 3.2.2 bereits erklärt wurde, kann Cloud am Beispiel von Platooning hauptsächlich zur Vor- und Nachbereitung genutzt werden. Eine der Hauptaufgaben stellt hierbei die Suche von verfügbaren Zügen dar, mit welchem sich das jeweilige Fahrzeug koppeln kann. Hierzu liegt ein großer Datensatz aus bestehenden Zügen vor, aus welchem der jeweils ideale zu ermitteln gilt.

4.1 Ziele und Kriterien

Das Ziel dieser prototypischen Umsetzung ist es zu untersuchen, inwiefern der Einsatz der Cloud für den Anwendungsfall V2X am Beispiel Platooning sinnvoll ist. Um dieses Ziel zu erreichen und ein aussagekräftiges Ergebnis zu erhalten, wird die Effizienz in Anbetracht der Zeit, die Richtigkeit der jeweiligen Ausgabe und die dabei anfallenden Kosten gemessen und beurteilt. Hierzu wird anhand von unterschiedlichen Datensätzen die Suche nach einem passenden Zug, das Hinzufügen eines neuen Zuges, das Ausgeben eines konkreten Zuges und das Auflösen bzw. Entfernen eines Zuges simuliert und getestet.

Anhand der Ergebnisse dieser Umsetzung und der Kriterien und Anforderungen an die Cloud für den Anwendungsfall V2X, welche in 3.2.2 erarbeitet wurden, soll dieses Ziel erreicht werden.

4.2 Architektur

Für diese Umsetzung wird eine Cloud-Datenbank benötigt, in welcher die jeweiligen Datensätze gelagert werden. Hierzu muss zunächst eine Cloud-Datenbank aufgesetzt und für den Anwendungsfall passende Datensätze generiert werden. Diese Datensätze müssen anschließend in der Cloud gespeichert und abgerufen werden können.

4.2.1 Clouddienste in GCP

Für diesen Anwendungsfall wurde sich für die Google Cloud Platform (GCP) entschieden, welche neben AWS und Microsoft Azure zu dem größten Cloud-Anbieter zählt. Der Grund dafür ist, dass GCP durch seine Schnittstellen zu den anderen Google-Diensten sehr benutzerfreundlich ist, dank des Startguthabens die Kalkulation der Anfallenden Kosten besser überschaut werden können und der Dienst nicht nur mithilfe einer Kreditkarte, sondern auch über Paypal genutzt werden kann. GCP setzt sich zusammen aus einer Reihe von Cloud-Computing-Diensten, welche auf derselben Infrastruktur laufen wie die Endbenutzerprodukte „Google Suche“ und „Youtube“ von Google selbst. Die wichtigsten Cloud Datenbank und Speicherdienste von GCP sind Big Query, Cloud SQL und Cloud Storage [92] [95].

- Big Query: Ist ein Data Warehouse, welches durch schnelle SQL-Abfragen das Speichern und Abfragen umfangreicher Datasets ermöglicht. Es wurde konzipiert für sehr große Datenmengen, schnelle Analysen und verfügt über integriertes maschinelles Lernen.
- Cloud SQL: Ist ein Dienst für relationale Datenbanken für MySQL, PostgreSQL und SQL-Server, welcher eine hohe Performance und kurze Antwortzeiten bietet.
- Google Cloud Storage: Hierbei handelt es sich um einen Objektspeicher, welcher die Speicherung und das Abrufen von beliebigen Datenmengen ermöglicht. Cloud Storage besitzt demnach kein Schema, aber ermöglicht dennoch SQL-Abfragen.

Im Zuge dieses Anwendungsfalls wurde sich für den Cloud SQL-Service entschieden, da die Daten über bestehende und verfügbare Züge in Form einer Tabelle verwaltet werden sollen und möglichst kurze Antwortzeiten notwendig sind.

Um die Cloud SQL einzurichten, benötigt man ein Google-Konto und zudem muss eine Zahlungsart bei Google selbst hinterlegen sein. Anschließend erhält man ein Guthaben von 300 \$, mit welchen die über 20 Produkte genutzt werden können. Damit wurde die Cloud SQL gestartet und eine Datenbank für das Platooning erstellt.

4.2.2 Testdaten

Die Testdaten, welche die einzelnen Züge des Platooning darstellen, wurden in einer Google Tabelle gesammelt. Bei Google Tabellen handelt es sich um ein Tabellenkalkulationsprogramm, welches wie Excel Daten speichern und auswerten kann. Die Daten können dabei von jedem Ort aus aufgerufen und bearbeitet werden. Durch die Schnittstellen und Benutzerfreundlichkeit der GCP ist es möglich, die einzelnen Dienste und Services von Google zu verbinden und die Tabelle somit in die Datenbank zu übertragen. Zudem wurden die Werte mit Formeln generiert. Durch einen Klick lassen sich diese zufällig neu generieren und verändern, wodurch ein aussagekräftigeres Ergebnis erzielt werden kann. Somit wurden schlussendlich 500.000 unterschiedliche Datensätze generiert. Die Liste mit den Datensätzen ist im Anhang vorhanden.

Wie in Abbildung 18 dargestellt, besitzt jeder Datensatz Informationen über seine ID, die Position des Zuges, dessen aktuelle und maximale Länge, seine Höchstgeschwindigkeit und das Ziel. Alle weiteren Informationen sind variabel und durch Formeln veränderbar. Mithilfe einer externen Schnittstelle wurde die Plattform „holistics“ [96] verwendet, um die Tabelle automatisch zu aktualisieren, sofern sich die Daten in der Tabelle ändern. Hierzu musste lediglich deren Netzwerk und die eigene IP in der GCP autorisiert werden.

id	pos	current_vehicle	max_vehicle	top_speed	city
1	12417	2	4	80	A
2	82002	2	5	90	E
3	86709	5	5	100	E
4	98515	2	4	100	F
5	27296	3	3	90	C
6	76217	2	2	90	D

Abbildung 18: Platooning-Testdaten

4.3 Durchführung

Das Ziel des Anwendungsfalls ist es, anhand der Testdaten und der Parameter des eigenen Fahrzeuges einen passenden Zug für das eigene Fahrzeug zu ermitteln. Mittels dieser Durchführung und der daraus resultierenden Informationen, welche die Abfragezeit und nötigen Rechenleistung beinhalten, soll anschließend das zuvor in 4.1 definierte Ziel erreicht werden. Um dies vereinfacht darstellen zu können, wurden die Positionen der Züge, wie in Abbildung 19 dargestellt, entlang einer Strecke definiert. Der Pkw stellt dabei das Fahrzeug dar, welches sich einem Zug anschließen möchte, wohingegen die Lkw einzelne Züge repräsentieren, deren Daten in der Cloud-Tabelle gespeichert sind.

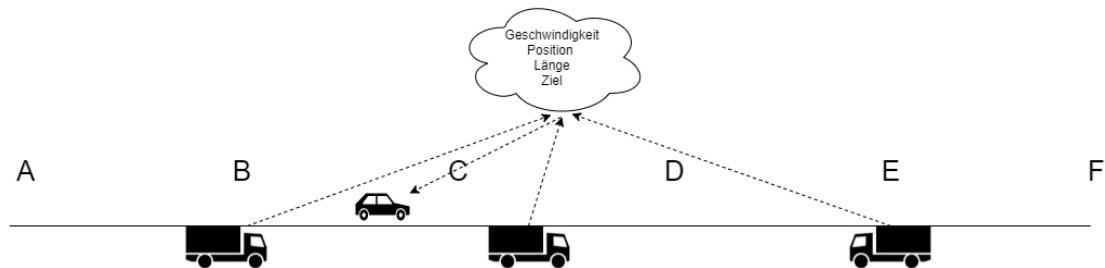


Abbildung 19: Darstellung der Züge auf einer vorgegebenen Strecke

Damit der Pkw den optimalen Zug ermitteln kann, müssen Kriterien festgelegt werden, welche die jeweilige Eignung verschiedener Züge beurteilt. Zur Visualisierung wurde ein Entscheidungsbaum erstellt (Abbildung 20). Mithilfe der eigenen Informationen kann dadurch festgestellt werden, ob ein Zug geeignet oder ungeeignet ist. Als besonders ungeeignet stellten sich Züge heraus, welche:

- die maximale Zuglänge erreicht haben
- ein Ziel ansteuern, welches nicht auf dem Weg des eigenen Ziels liegt und nicht mit dem eigenen Ziel identisch ist
- sich zu weit entfernt befinden
- eine höhere Geschwindigkeit fahren, als es das eigene Fahrzeug zulässt.

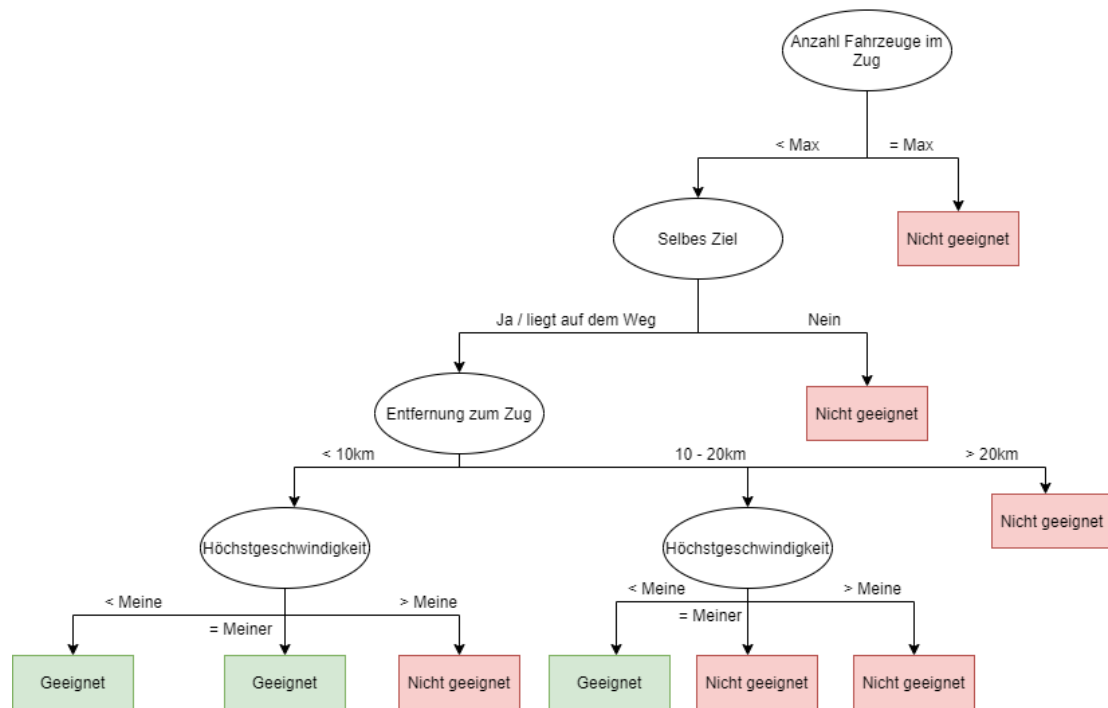


Abbildung 20: Entscheidungsbaum zur Eignung verschiedener Züge

Anschließend wurde ein passendes SQL-Statement erstellt, welches diesen Entscheidungsbaum widerspiegelt (Abbildung 21). Dabei werden Züge mit kürzerer Länge und gleicher Geschwindigkeit den anderen bevorzugt, damit sich zum einen die kürzeren Züge schneller füllen können und damit zum anderen das Risiko für Unfälle und die zu berechnenden Manöver geringer sind. Aus der ganzen Liste an Daten werden die zehn geeignetsten Züge herausgefiltert und absteigend ausgegeben.

```
4 • SELECT *
5 FROM platooning.platooning_data2
6 WHERE current_vehicle != max_vehicle #Zug ist nicht voll
7 AND top_speed <= @myTopSpeed #Sind nicht schneller als mein Max
8 AND city = @targetCity #Zug hat selbes Ziel
9 AND ABS( pos - @myPos ) < 20 #Zug ist nicht weiter als 20 entfernt
10 ORDER BY ABS( pos - @myPos ),
11 top_speed DESC,
12 current_vehicle ASC LIMIT 10; #Sortieren nach nächsten und schnelleren Züge
```

Abbildung 21: SQL-Statement zur Ermittlung eines passenden Zuges

4.4 Ergebnisse und Bewertung

Die Abfrage aus Abbildung 21 wurde für folgenden Variablen getätigt:

- @myPos = 65930
- @targetCity = „D“
- @myTopSpeed = 90.

Mit den aktuellen Daten der Tabelle ergab sich dadurch folgendes Ergebnis (Abbildung 22):

id	pos	current_vehicle	max_vehicle	top_speed	city
311827	65931	1	2	90	D
357682	65929	1	5	90	D
373892	65931	3	4	90	D
302266	65931	1	5	80	D
7262	65932	1	5	90	D
238118	65928	1	3	90	D
155683	65928	2	3	90	D
30259	65928	1	2	80	D
155976	65932	1	3	80	D
408750	65933	3	4	90	D

Abbildung 22: Ergebnis für die Suche nach einem passenden Zug

Neben dem Finden der idealen Züge wurde das Hinzufügen eines neuen Zuges in die Liste, das Ausgeben eines spezifischen Zuges aus der Liste und das Entfernen eines Zuges getestet, wobei

nach tausend Wiederholungen mit verschiedenen Lasten folgende durchschnittliche Antwortzeiten festgestellt wurden:

Tabelle 6: Abfrageergebnisse

Abfrage	Durchschnittliche Abfragezeit		
	100.000	250.000	500.000
Datensätze			
Passenden Zug finden	0,094 sek	0,200 sek	0,391 sek
Neuen Zug hinzufügen	0,020 sek	0,024 sek	0,032 sek
Ausgeben eines bestimmten Zuges	0,078 sek	0,172 sek	0,344 sek
Zug aus der Liste entfernen	0,086 sek	0,195 sek	0,375 sek

Alle vier Abfragen wurden zuverlässig und mit einer durchschnittlich geringen Latenz durchgeführt. Zu beachten ist dennoch, dass die Abfragen hierbei unter Idealbedingungen an einem lokalen Rechner durchgeführt wurden. Im Falle des Platooning würden diese Abfragen von verschiedenen Fahrzeugen über dessen Systeme durchgeführt werden, welche keine kontinuierliche Verbindung zum Internet garantieren können und sich darüber hinaus in Bewegung befinden. Ein weiterer Störfaktor wäre zudem der Datenverkehr, welcher durch alle Fahrzeuge und Züge verursacht wird.





Nichtsdestotrotz liegt die durchschnittliche Antwortzeit zum Finden eines passenden Zuges, zum Ausgeben eines gezielten Zuges und zum Entfernen eines Zuges bei 500.000 Datensätzen unter idealen Bedingungen bei etwa 0,4 Sekunden (400 ms). Das Erstellen eines Zuges geht hierbei in unter 0,05 Sekunden (50 ms). Somit wird unter aktuellen gegebenenheiten keine Echtzeitverarbeitung der Anfrage erreicht. Wie in 3.2.2 erwähnt wird der Einsatz der Cloud, am Beispiel vom Platooning, in der Vor- und Nachbereitung liegen. Hierfür sind weniger Echtzeitverarbeitungen, aber dafür mehr Zuverlässige Verarbeitungen in kurzer Zeit notwendig. Eine Gegenprüfung der Ergebnisse hat ergeben, dass diese richtig und im Rahmen der benötigten Zeit zuverlässig wiedergegeben wurden, weshalb dieses Kriterium durch die Cloud erfüllt wird. Somit wurde bewiesen, dass die Cloud für dieses Aufgabengebiet geeignet ist.

Als weiterer Punkt wurden die Kosten untersucht, welche sich daraus ergaben. Eine Berechnung der Kosten unter den aktuellen Bedingungen hat ergeben, dass sich diese auf 649 \$ im Monat belaufen würden. Diese Kosten ergeben sich für den Server in Frankfurt, welcher eine Hochverfügbarkeit, ein SSD Storage von 100 Gigabyte, 26 Gigabyte RAM und vier vCPUs besitzt. Im Falle dieses Prototyps wurden von den 100 Gigabyte Speicherplatz lediglich 10 Gigabyte verwendet und zusätzliche Kosten sind aufgrund des Startguthabens von 300 \$ nicht angefallen. Nichtsdestotrotz würde diese Leistung nicht ausreichen, um ein System wie das Platooning unter realen Bedingungen durchzuführen. Die CPU ist durch geringe Wiederholung der Anfrage schon an ihre Grenzen gestoßen. Dabei wurde die Suche nach einem idealen Zug von 10 Instanzen simuliert. Demnach wurde gezeigt, dass eine Standardausstattung für den realen Betrieb nicht ausreichen wird. Neben der Standardausstattung kann eine höhere und geringere Rechenleistung gewählt werden. Die Kosten für die höchste Ausstattung mit 16 vCPUs und 104 Gigabyte RAM würde monatlich 2,473 \$ betragen, wohingegen die leichteste Ausstattung mit einer vCPU und 3,75 Gigabyte RAM sich monatlich auf 159 \$ belaufen würde (Abbildung 23). Um Platooning mit wenigen Fahrzeugen im realen Straßenverkehr einsetzen zu können, wird mindestens die höchste Ausstattung der Rechenleistung benötigt werden.

Die Kosten für eigene Inhouse-Server wären bei diesen Vorhaben deutlich höher, da sich diese nicht ausschließlich auf die Nutzung, sondern zusätzlich auf Betriebs- und Instandhaltungskosten belaufen würden. Außerdem wäre eine einfache Skalierung der Leistung in diesem Falle nicht möglich. Eine Skalierung kann in der Cloud variabel und auch zeitbedingt stattfinden. Somit wird beispielsweise in Deutschland an Sonn- und Feiertagen voraussichtlich weniger Gebrauch vom Platooning gemacht, da ein Fahrverbot für viele Lastkraftwagen besteht. Andererseits kann dieser Bedarf wiederum am Folgetag mit zunehmendem Güterverkehr steigen. In der Cloud kann dies bedarfsgerecht eingeteilt und angepasst werden, wodurch sich die Cloud im Bereich der Kosten als lukrativer erweist.

Im gesamten wurde deshalb bewiesen, dass die Entwicklung und der Betrieb des Platooning durch die Cloud sowohl im Anbetracht der anfallenden Kosten als auch seiner Zuverlässigkeit als sinnvoll zu betrachten ist und Inhouse-Servern vorgezogen werden sollte.

Prototypische Umsetzung

DB-LIGHTWEIGHT-1 HA  	DB-HIGHMEM-16 HA  
# of instances: 1	# of instances: 1
Instance type: db-lightweight-1	Instance type: db-highmem-16
Location: Frankfurt	Location: Frankfurt
730.0 total hours per month	730.0 total hours per month
SSD Storage: 100.0 GiB	SSD Storage: 100.0 GiB
Backup: 0.0 GiB	Backup: 0.0 GiB
USD 159.15	USD 2,473.98

a) b)

Abbildung 23: Preiskalkulation für a) die leichteste und b) die höchste Ausstattung [97]

5 Fazit und Ausblick

In der vorliegenden Arbeit wurde sich mit der Frage beschäftigt, welches Potenzial die Cloud für die ADAS/AD-Entwicklung bereithält. Dazu wurden speziell die Anwendungsbereiche zur Entwicklung von HD-Karten, die V2X-Technologie am Beispiel vom Platooning und die Vision des Smart Parking analysiert. Hierbei wurden die einzelnen Anwendungsbereiche hinsichtlich ihrer Funktion beschrieben und anschließend auf aktuelle Probleme und Maßnahmen zur Beseitigung dieser untersucht. Anschließend erfolgte auf Grundlage der erarbeiteten Erkenntnisse die prototypische Umsetzung für das Platooning. Dazu wurde die Google Cloud Platform mit deren Cloud SQL-Service verwendet, um anhand von bis zu 500.000 Datensätzen die Latenz und Rechenfunktionen zu prüfen und erste Einblicke in die Kosten zu erhalten.

Die Ergebnisse zeigen, dass der Einsatz der Cloud nicht für sicherheitskritische Aufgaben, welche auf eine Echtzeitverarbeitung angewiesen sind, geeignet ist. Die Cloud-Server befinden sich meistens geografischen sehr weit von den Endgeräten entfernt und eine stabile Verbindung zu diesen Servern kann aufgrund der Geschwindigkeit der Fahrzeuge und des mobilen Internets in vielen Ländern nicht zu jeder Zeit gewährleistet werden. Die Latenz wäre dabei zu hoch und das Fahrzeug könnte nicht in angemessener Zeit reagieren, wodurch auch die Entwicklung beeinträchtigt wäre. Auch die Ergebnisse der prototypischen Umsetzung zeigen, dass die Latenz selbst unter idealen Bedingungen mit bis zu 400 ms zu hoch ist und keine Echtzeitübertragung und -verarbeitung ermöglicht.

Anders jedoch verhält es sich mit rechenintensiven Aufgaben, welche zur Vor- und Nachbereitung, zur Analyse oder auch gemeinsamen Nutzung dienen und keine direkte Echtzeitverarbeitung benötigen. Hierbei wurde gezeigt, dass sich der Einsatz der Cloud positiv auf den Verlauf der Entwicklung auswirken würde. Hybride Clouds würden die Zusammenarbeit zwischen Unternehmen und Organisationen begünstigen. Vorhandene Kartendaten und neu gesammelte Daten von normalen Fahrzeugen können über Schnittstellen in das eigene System integriert

werden und eine eigene Kartierung ersparen. Außerdem wurde mit dem Prototyp demonstriert, dass die Vorteile der Cloud bezüglich der Skalierung und Rechenfunktionen genutzt werden können, um die dynamische Anzahl an Nutzern zuverlässig bewältigen zu können, ohne selbst große Anschaffungen tätigen zu müssen.

5.1 Ausblick

Diese Arbeit hat veranschaulicht, dass eine Echtzeit-Übertragung von großen Datenmengen mit der Cloud allein meistens nicht möglich ist. Als eine Lösung zur Bekämpfung von großen Latenzen und weiteren sinnvollen Nutzung der Cloud wurde deshalb der Einsatz von Edge-Computing in 3.2.2 vorgeschlagen. Beim Edge-Computing handelt es sich um eine dezentrale Verarbeitung von Daten. Die Server befinden sich dabei am Rand des Netzwerkes und werden nicht auf zentralen Servern gesammelt, sondern an lokalen Knotenpunkten bearbeitet. Dadurch wird es möglich, Aufgaben schneller zu bearbeiten und somit für geringere Latenz zu sorgen. Dennoch befinden sich global nicht immer verfügbare Server. In diesen Zusammenhang wäre es interessant zu untersuchen, wie sich diese in Zusammenarbeit mit der Cloud und modernen Fahrzeugen auf dessen Entwicklung auswirken. Dabei könnte vor allem eine sinnvolle Ressourcenverteilung zwischen den beiden Servern, die Auswirkung auf die Latenz und Möglichkeiten zur Skalierung und Verfügbarkeit untersucht werden. Zu Anbietern, welche in Edge-Computing investieren, gehören unter anderem IBM, Siemens und Bosch.

Des Weiteren wäre eine Untersuchung des in Kapitel 4 entworfenen Prototyps unter realen Bedingungen nötig, wobei ein mit dem Internet verbundenes Fahrzeug im Straßenverkehr auf die Cloud-Plattform zugreift und mithilfe seiner Positions- und Fahrzeugdaten aus der gegebenen Liste den Passenden Zug heraussucht. Untersucht werden könnte dabei die tatsächliche Latenz und Last durch das Fahrzeug.

Auch Anwendungen und Prototypen bezogen auf die anderen beiden Anwendungsfälle könnten untersucht und umgesetzt werden.

Literaturverzeichnis

- [1] S. Jursch, „PwC,“ [Online]. Available: <https://www.pwc.de/de/nachhaltigkeit/nachhaltigkeit-in-der-automobilindustrie.html>. [Zugriff am 28 Mai 2021].
- [2] D. G. u. M. S. J. Dargay, „Vehicle ownership and income growth, worldwide: 1960-2030,“ *The Energy Journal*, 2007, pp. 143-170.
- [3] J. D. O. a. J. J. Y. C. Olariu, „The role auf cloud-computing in the development and application of ADAS,“ *European Signal Processing Conference*, 2018, pp. 1037-1041.
- [4] „eMove360,“ 22 Juni 2020. [Online]. Available: <https://www.emove360.com/de/entwicklung-autonomer-fahrzeuge-in-der-cloud/>. [Zugriff am 28 Mai 2021].
- [5] T. Nguyen, „Autonomes Fahren beginnt im Rechenzentrum,“ *Digitale Welt 4*, pp. 34-37, 2020.
- [6] D. Dahlmann, „Grunderszene,“ 18 Februar 2021. [Online]. Available: <https://www.businessinsider.de/gruenderszene/automotive-mobility/drehmoment-durchbruch-autonomes-fahren-r2/>. [Zugriff am 28 Mai 2021].

- [7] S. Schaal, „Wie die Cloud das Auto revolutioniert,“ 07 Juli 2015. [Online]. Available: <https://www.wiwo.de/technologie/vernetztes-fahren-wie-die-cloud-das-auto-revolutioniert/11996022.html>. [Zugriff am 15 Juli 2021].
- [8] „Die Bundesregierung,“ 28 Mai 2021. [Online]. Available: <https://www.bundesregierung.de/breg-de/suche/faq-autonomes-fahren-1852070>. [Zugriff am 15 Juli 2021].
- [9] „THE HANSEN REPORT On Automotive Electronics,“ *ATZ electronics*, pp. 24-26, 2020.
- [10] H. Schuster, „IT-Business,“ 13 Mai 2015. [Online]. Available: <https://www.it-business.de/anforderungen-an-die-it-ueberfordern-a-489161/>. [Zugriff am 16 Juli 2021].
- [11] „Statista,“ 2021. [Online]. Available: <https://de.statista.com/statistik/daten/studie/177484/umfrage/einsatz-von-cloud-computing-in-deutschen-unternehmen-2011/>. [Zugriff am 16 Juli 2021].
- [12] J. Repschläger, D. Pannicke und R. Zarnekow, „Cloud Computing: Definition, Geschäftsmodelle und Entwicklungspotenziale,“ *HDM Praxis der Wirtschaftsinformatik*, pp. 6-15, 2010.
- [13] „Global cloud infrastructure market Q3 2020,“ Canalys, 29 Oktober 2020. [Online]. Available: <https://www.canalys.com/newsroom/worldwide-cloud-market-q320>. [Zugriff am 16 Juli 2021].
- [14] E. Jones, „AWS vs Azure im Jahr 2021 (Vergleich der Cloud Computing Giganten),“ Kinsta, 25 März 2021. [Online]. Available: <https://kinsta.com/de/blog/aws-vs-azure/>. [Zugriff am 16 Juli 2021].

- [15] C. Braun, „Cloud-, Grid-, Cluster- und Meta-Computing,“ 2008, pp. 132-133.
- [16] P. Mell und T. Grance, „The NIST Definition of Cloud,“ NIST, 2011.
- [17] „Wie Cloud Transformation Geschäftsmodelle Digital, Agil, Skalierbar und Kundenzentriert macht,“ Defacto, 25 Mai 2021. [Online]. Available: <https://www.defacto.de/wie-cloud-transformation-geschaeftsmodelle-digital-agil-skalierbar-und-kundenzentriert-macht/>. [Zugriff am 19 Juli 2021].
- [18] C. Weinhardt, A. Anandasivam, B. Blau, N. Borissov, T. Meinl, W. Michalk und J. Stößer, „Cloud-Computing – Eine Abgrenzung, Geschäftsmodelle und Forschungsgebiete,“ in *WI – State of the art*, Gabler Verlag, 2009, pp. 453-462.
- [19] „Statista,“ 2021. [Online]. Available: <https://de.statista.com/>. [Zugriff am 17 Juli 2021].
- [20] „techopedia,“ [Online]. Available: <https://www.techopedia.com/definition/14469/measured-service-cloud-computing>. [Zugriff am 17 Juli 2021].
- [21] P. H. Deussen, L. Strick und J. Peters, „Cloud-Computing für die öffentliche Verwaltung,“ 2021, pp. 15-24.
- [22] W. Dauschek, „techbold,“ 26 Juli 2018. [Online]. Available: <https://www.techbold.at/blog/cloud-services-und-dienste-einfach-erklart>. [Zugriff am 18 Juni 2021].
- [23] E. Jones, „Arten von Cloud Computing – ein umfangreicher Leitfaden zu Cloud-Lösungen und -Technologien im Jahr 2021,“ Kinsta, 21 August 2020. [Online]. Available: <https://kinsta.com/de/blog/arten-von-cloud-computing/>. [Zugriff am 18 Juli 2021].

- [24] bmc, [Online]. Available: <https://www.bmc.com/>. [Zugriff am 19 Juli 2021].
- [25] U. Winkelhake, Die digitale Transformation der Automobilindustrie, Springer Vieweg, 2017.
- [26] M. Böttger, Cloud Computing richtig gemacht: Ein Vorgehensmodell zur Auswahl von SaaSAnwendungen, Diplomica Verlag, 2012.
- [27] M. Maurer, J. C. Gerdes, B. Lenz und H. Winner, Autonomes Fahren: Technische, rechtliche und gesellschaftliche Aspekte, Springer Vieweg, Berlin, Heidelberg, 2015.
- [28] Q. L. D. T. Y. L. H. K. a. S. S. H. Lu, „The Cognitive Internet of Vehicles for Autonomous Driving,“ IEEE Network, 2019, pp. 65-73.
- [29] „BMVI,“ [Online]. Available: <https://www.bmvi.de>. [Zugriff am 21 Juli 2021].
- [30] J. Mazzega, „Pegasus Method: an Overview,“ 2019. [Online]. Available: <https://www.pegasusprojekt.de/de/>. [Zugriff am 21 Juli 2021].
- [31] C. Ott, „Eine durchgehende Datenpipeline beschleunigt den Fortschritt,“ Computerworld, 21 Juli 2020. [Online]. Available: <https://www.computerworld.ch/technik/firmenbeitraege/durchgehende-datenpipeline-beschleunigt-fortschritt-2556072.html>. [Zugriff am 26 Juli 2021].
- [32] „Autonomes Fahren: Die 5 Stufen zum selbstfahrenden Auto,“ ADAC, 07 November 2018. [Online]. Available: <https://www.adac.de/rund-ums-fahrzeug/ausstattung-technik-zubehoer/autonomes-fahren/grundlagen/autonomes-fahren-5-stufen/>. [Zugriff am 21 Juli 2021].

- [33] „Die 5 Level des Autonomen Fahrens,“ BMW, 21 Oktober 2020. [Online]. Available: <https://www.bmw.com/de/automotive-life/autonomes-fahren.html>. [Zugriff am 21 Juli 2021].
- [34] P. Nickel und T. Schöpfner, „Sensoren ersetzen Sinne - ADAS-Datenerfassung für die Funktionsentwicklung,“ *ATZ Elektron*, pp. 50-53, 2021.
- [35] K. Leicht, „Die Stufen des autonomen Fahrens,“ TÜV-Nord, 24 Januar 2019. [Online]. Available: <https://www.tuev-nord.de/explore/de/erklaert/die-stufen-des-autonomen-fahrens/>. [Zugriff am 21 Juli 2021].
- [36] J. Hampel, C. Kropp und M. M. Zwick, „Zur gesellschaftlichen Wahrnehmung des voll autonomen Fahrens und seiner möglichen nachhaltigkeitsbezogenen Implikationen,“ *TATuP*, pp. 38-45, 2018.
- [37] H. G. Seif und X. Hu, „Autonomous Driving in the iCity—HD Maps as a Key Challenge of the Automotive Industry,“ *Engineering*, pp. 159-162, 2016.
- [38] W. W. und W. H., „Die Freigabe des autonomen Fahrens,“ in *Autonomes Fahren*, Springer Vieweg, Berlin, Heidelberg, 2015, pp. 439-464.
- [39] F. Schuldt, T. Menzel und M. Maurer, „Eine Methode für die Zuordnung von Testfällen für automatisierte Fahrfunktionen auf X-in-the-Loop Verfahren im modularen virtuellen Testbaukasten,“ 2015. [Online]. Available: <http://docplayer.org/7071323-Eine-methode-fuer-die-zuordnung-von-testfaellen-fuer-automatisierte-fahrfunktionen-auf-x-in-the-loop-verfahren-im-modularen-virtuellen-testbaukasten.html>. [Zugriff am 23 Juli 2021].
- [40] J. Wagner und J. Häring, „Validierung von hochautomatisierten Fahrfunktionen mit cloud-basierter Simulation,“ *ATZ Automobiltech*, pp. 50-56, Januar 2020.

- [41] S. Thrun, „Vom autonomen Fahrzeug zum Flugtaxi,“ *dSPACE Magazin*, pp. 50-58, Februar 2019.
- [42] H. Kagalwala, S. Srivastava, M. Venkatesan und S. Srinivasan, „Implementation Methodologies for Simulation as a Service (SaaS) to Develop ADAS Applications,“ *SAE International*, Januar 2021.
- [43] „Hardware-in-the-Loop-Validierung elektronischer Steuereinheiten,“ Fraunhofer, [Online]. Available: <https://www.itwm.fraunhofer.de/de/abteilungen/sys/maschinenmonitoring-und-regelung/hardware-in-the-loop.html>. [Zugriff am 24 Juli 2021].
- [44] B. Holzinger, „Schnellere Tests von Radar- und anderen Sensoren für ADAS,“ all-electronics, 17 Mai 2021. [Online]. Available: <https://www.all-electronics.de/elektronik-entwicklung/schnellere-tests-von-radar-und-anderen-sensoren-fuer-ad-as-systeme-130.html>. [Zugriff am 24 Juli 2021].
- [45] J. Wei, „Hybrid mobile computing for connected autonomous vehicles,“ 2018.
- [46] V. Johanning und R. Mildner, *Car IT kompakt: Das Auto der Zukunft - Vernetzt und autonom fahren*, Springer Vieweg, 2015.
- [47] L. Shaoshan, T. Jie, W. Chao, W. Quan und G. Jean-Luc, „Implementing a Cloud Platform for Autonomous Driving,“ 2017. [Online]. Available: <https://arxiv.org/ftp/arxiv/papers/1704/1704.02696.pdf>. [Zugriff am 25 Juli 2021].
- [48] K. Reif, *Sensoren im Kraftfahrzeug*, Springer Vieweg, 2016.
- [49] P. Schiekofer, „Aufbau eines Technologieträgers für das autonome und elektrische Fahren,“ *ATZ Automobiltech*, pp. 54-59, 2020.

- [50] H. W., Y. G. und X. L. D., „Developing Vehicular Data Cloud Services in the IoT Environment,“ *IEEE Transactions on Industrial Informatics*, pp. 1587-1595, Mai 2014.
- [51] D. Zwick, „Volkswagen zapft seinen Datenschutz an,“ WELT, 13 Juli 2021. [Online]. Available: <https://amp.welt.de/wirtschaft/article231770701/Autonomes-Fahren-Volkswagen-zapft-seinen-Datenschutz-an.html>. [Zugriff am 26 Juli 2021].
- [52] S. Rossbach, „doubleSlash,“ 13 Februar 2020. [Online]. Available: <https://blog.doubleslash.de/hochautomatisiertes-fahren-findet-auch-in-der-cloud-statt/>. [Zugriff am 27 Juni 2021].
- [53] „now.digital,“ 26 März 2021. [Online]. Available: <https://now.digital/best-practice-digital-automotive/volkswagen-group-teams-up-with-microsoft-to-develop-automated-driving/>. [Zugriff am 19 Juli 2021].
- [54] C. Velten und S. Janata, „Cloud Computing,“ Januar 2012. [Online]. Available: http://www.isis-specials.de/profile_pdf/1e226_ed_saas0212.pdf. [Zugriff am 27 Juni 2021].
- [55] R. Isermann, *Fahrerassistenzsysteme 2017*, Springer Vieweg, 2017.
- [56] bussgeldkatalog.org, *Mobilitätsmagazin*, 14 Mai 2021. [Online]. Available: <https://www.bussgeldkatalog.org/lidar/>. [Zugriff am 19 Juli 2021].
- [57] „NVIDIA GeForce GTX 480 (GF100 / Fermi) im Performance Test,“ *tweakpc*, [Online]. Available: https://www.tweakpc.de/hardware/tests/grafikkarten/nvidia_geforce_gtx_480/s06.php. [Zugriff am 19 Juli 2021].

- [58] „Elektrische Konsumenten im Auto – soviel Strom verbrauchen sie,“ VARTA Battery World, [Online]. Available: <https://batteryworld.varta-automotive.com/de-de/elektrische-konsumenten-im-auto-soviel-strom-verbrauchen-sie>. [Zugriff am 19 Juli 2021].
- [59] „Mobilfunkstandards 2G, 3G, 4G und 5G - Alle Infos zu den Generationen,“ Sparhandy, [Online]. Available: <https://www.sparhandy.de/mobiles-internet/info/mobilfunkstandards/>. [Zugriff am 19 Juli 2021].
- [60] S. K., S. N., M. S. und N. A., „Vehicle control system coordinated between cloud and mobile edge computing,“ *55th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*, pp. 1122-1127, 2016.
- [61] W. Strijbosch, „Elektronik Praxis: Mit hochpräzisen HD-Karten wird autonomes Fahren machbar,“ 13 Oktober 2017. [Online]. Available: <https://www.elektronikpraxis.vogel.de/mit-hochpraezisen-hd-karten-wird-autonomes-fahren-machbar-a-652537/>. [Zugriff am 28 Juli 2021].
- [62] „TomTom,“ [Online]. Available: <https://www.tomtom.com/products/hd-map/>. [Zugriff am 28 Juli 2021].
- [63] „Here,“ [Online]. Available: <https://www.here.com/>. [Zugriff am 28 Juli 2021].
- [64] M. Linden, „16 Millionen Kilometer in Street View fotografiert,“ golem, 16 Dezember 2019. [Online]. Available: <https://www.golem.de/news/google-maps-16-millionen-kilometer-in-street-view-fotografiert-1912-145555.html>. [Zugriff am 29 Juli 2021].
- [65] B. Kahl, „Autonomous Driving Data Chain & Interfaces,“ Cornell University, 02 April 2021. [Online]. Available: <https://arxiv.org/abs/2104.01252>. [Zugriff am 30 Juli 2021].

- [66] „ZF,“ [Online]. Available: <https://www.zf.com/mobile/de/homepage/homepage.html>. [Zugriff am 01 August 2021].
- [67] J. Rabe, „Tesla Model 3: Unfall mit LKW/Truck,“ *Autobild*, 09 Juni 2020.
- [68] M. Nasimi, A. M. Habibi und H. Schotten, „Platoon–assisted Vehicular Cloud in VANET: Vision and Challenges,“ 25 August 2020. [Online]. Available: <https://arxiv.org/ftp/arxiv/papers/2008/2008.10928.pdf>. [Zugriff am 10 August 2021].
- [69] X. Boucherat, „V2V and the Cloud – essential for platooning,“ *Automotive World*, 09 August 2017. [Online]. Available: <https://www.automotiveworld.com/articles/v2v-cloud-essential-platooning/>. [Zugriff am 11 August 2021].
- [70] U. Montanaro, S. Fallah, M. Dianati, D. Oxtoby, T. Mizutani und A. Mouzakitis, „Cloud-Assisted Distributed Control System Architecture for Platooning,“ *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1258-1265, 2018.
- [71] D. Wang und T. Sun, „Leveraging 5G TSN in V2X Communication for Cloud Vehicle,“ *2020 IEEE International Conference on Edge Computing (EDGE)*, pp. 106-110, 2020.
- [72] T. Lavery, „dedicated short-range communication (DSRC),“ *WhatIS*, 2017. [Online]. Available: <https://whatis.techtarget.com/definition/dedicated-short-range-communication-DSRC>. [Zugriff am 14 August 2021].
- [73] „Dedicated Short Range Communications (DSRC) Service,“ *Federal Communications Commission*, 22 April 2019. [Online]. Available: <https://www.fcc.gov/wireless/bureau-divisions/mobility-division/dedicated-short-range-communications-dsrc-service>. [Zugriff am 14 August 2021].

- [74] C. Bernstein, „Cellular Vehicle to Everything (C-V2X),“ IoT Agenda, November 2018. [Online]. Available: <https://internetofthingsagenda.techtarget.com/definition/Cellular-Vehicle-to-Everything-C-V2X>. [Zugriff am 14 August 2021].
- [75] R. Lembke, „DSRC und C-V2C im Vergleich,“ SpringerProfessional, 06 März 2018. [Online]. Available: <https://www.springerprofessional.de/automatisiertes-fahren/automobilelektronik---software/dsrc-und-c-v2x-im-vergleich/15476434>. [Zugriff am 14 August 2021].
- [76] „How the Rise of Edge Computing will Reshape the Data Center Landscape,“ ConnectorSupplier, 08 Oktober 2019. [Online]. Available: <https://connectorsupplier.com/how-the-rise-of-edge-computing-will-reshape-the-data-center-landscape/>. [Zugriff am 17 August 2021].
- [77] FIS-Azubis, „Cloud Computing vs. Edge Computing,“ FIS, 02 Juli 2020. [Online]. Available: <https://www.fis-gmbh.de/de/blog/cloud-computing-vs-edge-computing/>. [Zugriff am 17 August 2021].
- [78] M. Hasan, M. Islam und Y. Alsaawy, „Smart Parking Model based on Internet of Things (IoT) and TensorFlow,“ *2019 7th International Conference on Smart Computing & Communications (ICSCC)*, pp. 1-5, 2019.
- [79] M. Müller, „Connected Parking: Eine Vision wird Wirklichkeit – willkommen in der neuen Welt des Parkens,“ Ernst & Sohn, Berlin, 2017.
- [80] F. Greis, „Community Based Parking: Mercedes S-Klasse liefert Daten für Boschs Parkplatzsuche,“ Golem, 02 August 2017. [Online]. Available: <https://www.golem.de/news/community-based-parking-mercedes-s-klasse-liefert-daten-fuer-boschs-parkplatzsuche-1708-129274.html>. [Zugriff am 31 August 2021].

- [81] R. Schlösser, „11.000 Sensoren und die Cloud für Hamburgs Autofahrer: Intelligent Parken mit NB-IoT,“ *Computerwoche*, 04 September 2017. [Online]. Available: <https://www.computerwoche.de/a/intelligent-parken-mit-nb-iot,3331487>. [Zugriff am 31 August 2021].
- [82] S. Mathur, S. Kaul, M. Gruteser und W. Trappe, „ParkNet: A Mobile Sensor Network for Harvesting Real Time Vehicular Parking Information,“ Mai 2009.
- [83] E. Cicek und S. Gören, „Fully automated roadside parking spot detection in real timewith deep learning,“ 14 September 2020. [Online]. Available: <https://onlinelibrary.wiley.com/doi/10.1002/cpe.6006>. [Zugriff am 31 August 2020].
- [84] M. Margreiter, F. Orfanou und P. Mayer, „Determination of the parking place availability using manual data collection enriched by crowdsourced in-vehicle data,“ *Transportation Research Procedia*, pp. 497-510, 2017.
- [85] „Community-based Parking,“ Bosch, 2021. [Online]. Available: <https://www.bosch-mobility-solutions.com/de/loesungen/parken/community-based-parking/>. [Zugriff am 05 September 2021].
- [86] „INRIX Bringt Erstmals On-Street-Parkdienste in Ein Vernetztes Automobil,“ INRIX, [Online]. Available: <https://inrix.com/>. [Zugriff am 05 September 2021].
- [87] M. Bosler, C. Jud und G. Herzwurm, „Connected-Car-Services: eine Klassifikation der Plattformen für das vernetzte Automobil,“ *HMD 54*, pp. 1005-1020, 2017.
- [88] „Making Sense of IoT Platforms: AWS vs Azure vs Google vs IBM vs Cisco,“ altexsoft, 20 Mai 2020. [Online]. Available: <https://www.altexsoft.com/blog/iot-platforms/>. [Zugriff am 08 September 2021].

- [89] P. Pierleoni, R. Concetti, A. Belli und L. Palma, „Amazon, Google and Microsoft Solutions for IoT: Architectures and a Performance Comparison,“ *IEEE Access*, pp. 5455-5470, 2020.
- [90] A. Muhammed und D. Ucuz, „Comparison of the IoT Platform Vendors, Microsoft Azure, Amazon Web Services, and Google Cloud, from Users’ Perspectives,“ *2020 8th International Symposium on Digital Forensics and Security (ISDFS)*, pp. 1-4, 2020.
- [91] „AWS IoT Core – Preise,“ AWS, 2021. [Online]. Available: <https://aws.amazon.com/de/iot-core/pricing/>. [Zugriff am 10 September 2021].
- [92] „Google Cloud Platform,“ Google Cloud, 2021. [Online]. Available: <https://cloud.google.com/>. [Zugriff am 10 September 2021].
- [93] K. Horne, „IoT Fight Night: AWS IoT Core vs Google Cloud IoT Core,“ NetBurner, 23 Oktober 2019. [Online]. Available: <https://www.netburner.com/learn/iot-cloud-aws-iot-core-vs-google-cloud-iot-core/>. [Zugriff am 10 September 2021].
- [94] „AWS Device Qualification Partners,“ AWS, 2021. [Online]. Available: <https://aws.amazon.com/de/partners/programs/dqp/partners/>. [Zugriff am 11 September 2021].
- [95] J. Klampfl, „Die wichtigsten Google Cloud Platform Datenbanken für die Webanalyse,“ e-dialog, 19 November 2020. [Online]. Available: <https://www.e-dialog.at/blog/webanalyse/die-wichtigsten-google-cloud-platform-datenbanken-fuer-die-webanalyse/>. [Zugriff am 22 September 2021].
- [96] „Modern Self-Service BI Platform,“ holistics, 2021. [Online]. Available: <https://www.holistics.io/>. [Zugriff am 04 Oktober 2021].

- [97] „Google Cloud Pricing Calculator,“ Google Cloud, [Online]. Available: <https://cloud.google.com/products/calculator#id=>. [Zugriff am 30 September 2021].

Anhang A: Datensätze zum Platooning

Die beigefügten Datensätze der einzelnen Züge des Platooning befinden sich auf der beigefügten CD im Verzeichnis:

/Anhang/Datensätze

Platooning_Daten.xlsx

Erklärung zur selbstständigen Bearbeitung einer Abschlussarbeit

Hiermit versichere ich, dass ich die vorliegende Arbeit ohne fremde Hilfe selbständig verfasst und nur die angegebenen Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich gemacht.

Ort

Datum

Unterschrift im Original